

版权注意事项：

- 1、书籍版权归作者和出版社所有
- 2、本PDF仅限用于个人获取知识，进行私底下的知识交流
- 3、PDF获得者不得在互联网上以任何目的进行传播
- 4、如觉得书籍内容很赞，请购买正版实体书，支持作者
- 5、请于下载PDF后24小时内删除本PDF。



微信公众平台 网页开发实战

HTML5+JSSDK混合开发解密

席新亮 等编著

公众号 + Web页面，让微信公众号更强大、更多彩
微信JSSDK + HTML5，让移动Web开发与微信结合轻而易举

本书从实战出发，揭秘微信公众平台网页开发
内容全面、案例丰富、轻松上手



中国工信出版集团



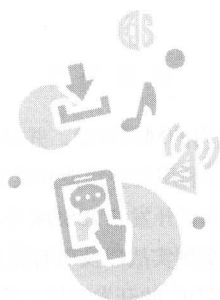
电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

席新亮



HTML5、JavaScript 技术爱好者，热衷于 HTML5、CSS3、Node.js 等前沿技术，写过 HTML5 游戏、搞过 Web，在实践中积累了大量经验。著有《超实用的 JavaScript 代码段》一书。新浪微博 @ 席新亮 HTML5，欢迎大家跟我交流。





微信公众平台 网页开发实战

HTML5+JSSDK混合开发解密

席新亮等编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

2017 年, 微信已经全面支持公众平台内的网页开发, 并且提供了网页开发样式库、JSSDK 库和网页内支付、网页调试工具, Web 开发人员可以利用这些功能开发出更多、更好、更炫的微信内应用。本书立志于帮助微信开发者更好地学习微信中的网页开发和 JSSDK 库。

本书涵盖了微信网页开发的所有 API 案例实战以及所有样式库。全书分为 13 章, 第 1~2 章介绍了不同类型微信公众号的区别和微信内网页 JSSDK 的开发环境; 第 3~9 章介绍了微信开放的所有 API, 并提供了最贴近现实的实例; 第 10 章通过实例演示了微信提供的网页样式; 第 11~12 章利用微信 JSSDK+HTML5 技术实现了两个大的项目应用; 最后一章介绍一些微信开发的技巧及问题。

本书适合所有微信开发人员使用, 尤其适合目前从事 Web 开发工作的人员。本书定位基础, 用案例结合 API 应用, 是目前市场上仅有的一本介绍微信网页结合 HTML5 开发的书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

微信公众平台网页开发实战: HTML5+JSSDK 混合开发解密 / 席新亮等编著. —北京: 电子工业出版社, 2017.5

ISBN 978-7-121-31363-9

I. ①微… II. ①席… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字 (2017) 第 077642 号

责任编辑: 董 英

印 刷: 三河市鑫金马印装有限公司

装 订: 三河市鑫金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱

邮编: 100036

开 本: 787×980 1/16

印张: 16

字数: 321 千字

版 次: 2017 年 5 月第 1 版

印 次: 2017 年 5 月第 1 次印刷

印 数: 4000 册 定价: 69.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888, 88258888。

质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: 010-51260888-819, faq@phei.com.cn。

前言 →

移动 Web 的发展速度令人惊叹，近几年所有互联网大企业都在往移动端发展，微信几乎占领了移动互联网的大部分江山，微信的公众号也已超过 1000 万，越来越多的企业不再建设网站，而是只要有一个包含所有功能的微信公众号就可以了，所有的服务行业也都创建了自己的公众号。由此可见，微信开发的人才需求非常旺盛。

微信公众号实际上就是一个 MINI 版的浏览器，可以说，HTML 5 的网页功能都能在微信公众号内部实现，甚至通过 JSSDK，HTML 5 网页还可以调用微信提供的所有接口，包括分享、扫一扫、支付、卡券、定位、照片、语音等功能。所以 Web 开发人员可以设计并实现更多的微信公众号内容，而不用学习更多移动开发的知识，只要会调用这些接口即可。

本书就是一本帮助读者学习微信公众平台+HTML 5 网页开发的入门书，只要有简单的 HTML 和 JavaScript 知识，就可以开发出一个功能完整的微信公众平台。

本书特色

为了与时俱进，从事前端开发工作的读者朋友，都应该了解、学习一下微信的 JSSDK，本书中介绍了微信 HTML5+JSSDK+微信网页样式的示例代码、技巧和实战案例。本书主要讲解形式如下：

- (1) 针对每个 API，都设计一个包含页面结构的完整效果，然后根据 API 中提供的各个接口，设计好针对每个接口的调用方法。

- (2) 所有的案例代码都很完整，读者在调试不通的情况下，可以直接使用本书资源中的源代码文件进行调试。

- (3) 很多人即使熟悉了 API 的接口，也不知道如何应用，本书最后还提供了两个完整

的项目案例，利用 JSSDK+HTML 5 的形式，把制作项目的流程、代码和技巧完整地演示出来。

本书的结构

第 1 章：由于做微信 HTML 5+JSSDK 开发离不开微信公众号，所以首先介绍了微信公众号的注册流程，并讲解一些关于微信后台的操作，以便于读者对微信后台有一个更清晰的认知。

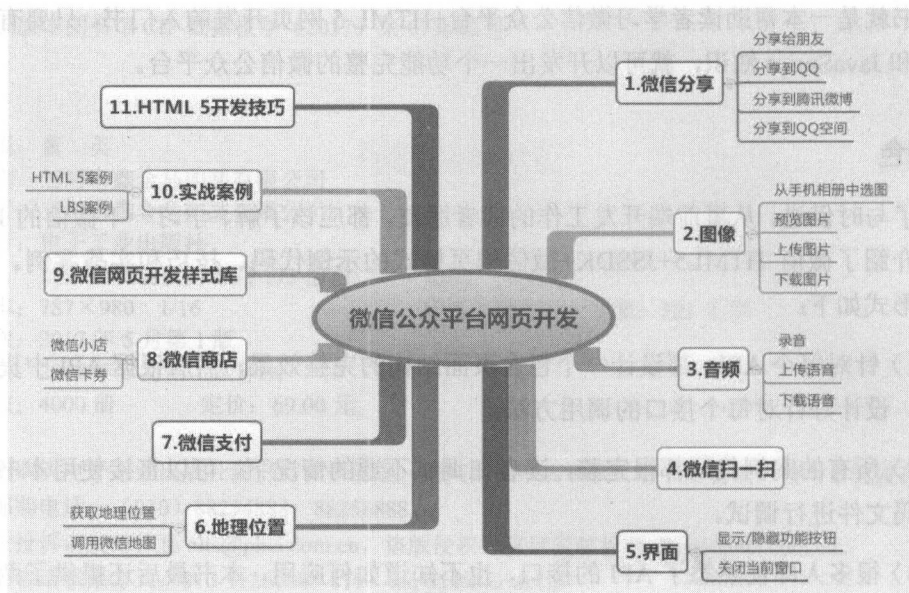
第 2 章：微信 JSSDK 的 API 使用需要搭建微信 JSSDK 的基础环境，本章综合讲解了搭建过程，为后续接口调用做铺垫。

第 3~9 章：阐述了微信开放的所有 API，包含代码示例、常见问题及一些小技巧。

第 10 章：详细介绍了如何使用微信提供的网页样式库，结合这些样式和 HTML 5，能够开发出更 Native 的微信公众号。

第 11~12 章：用案例作为载体，结合比较流行的 HTML 5 技术，讲解了微信 JSSDK 在实际项目中的重要作用。

第 13 章：总结了在微信 JSSDK 开发中可能面临的一些问题，随后又介绍了在微信调试模式下，一些提示信息的作用。



本书写给谁

- Web 网页开发人员
- PHP 开发人员
- 微信开发人员
- 大学毕业想利用微信平台创业的人员
- 微信公众号的维护人员
- 微信公众号的运营人员

由于作者个人能力有限，书中难免有疏漏之处，欢迎读者朋友提出宝贵的建议。

最后，感谢您购买此书，相信您也是一位前端技术爱好者。祝您在技术及职场生涯中，一帆风顺！

参与本书编写的还有：刘鑫、陈士领、陈丽、毛聪、王琳、张喆、薛淑英、李兰英、周洋、张学军、张兴瑜、陈宇、王健、张鑫，一并感谢他们。

席新亮

轻松注册成为博文视点社区用户（www.broadview.com.cn），您即可享受以下服务：

- 提交勘误：您对书中内容的修改意见可在【提交勘误】处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- 与作者交流：在页面下方【读者评论】处留下您的疑问或观点，与作者和其他读者一同学习交流。

页面入口：<http://www.broadview.com.cn/31363>



目 录

第 1 章 认识微信公众平台	1
1.1 微信公众平台账号分类	1
1.1.1 订阅号	2
1.1.2 服务号	2
1.1.3 企业号	4
1.1.4 小程序	4
1.2 微信公众号申请流程	5
1.3 微信公众号的认证	9
1.4 公众号设置	10
1.5 成为微信开发者	10
1.5.1 通过公众号	10
1.5.2 通过测试号	12
1.6 公众号的安全中心	13
1.7 本章小结	14
第 2 章 搭建网页 JSSDK 开发环境	15
2.1 JSSDK 介绍	15
2.2 部署 JSSDK	17
2.2.1 绑定域名	17
2.2.2 引入 JS 文件	18
2.2.3 通过 wx.config 接口验证 JSSDK 权限	18
2.3 使用 SAE 部署测试号的 JSSDK	28
2.3.1 在 SAE 中创建应用	28
2.3.2 上传测试代码	31

2.3.3	绑定 JS 接口	32
2.3.4	针对 SAE 不支持读写操作的处理——Memcached	34
2.3.5	在公众号中测试效果	37
2.4	JSSDK 的调试工具	38
2.5	JSSDK 的主要功能参考列表	38
2.6	一些其他基础入门必知的知识点	41
2.7	本章小结	41
第 3 章	JSSDK 网页开发的基础接口和分享接口	42
3.1	基础接口的作用	42
3.1.1	任何开发都要从基础接口开始	42
3.1.2	判断当前客户端版本是否支持指定 JS 接口	42
3.2	分享接口的作用	44
3.2.1	分享接口有什么好处	44
3.2.2	获取微信的分享接口	44
3.3	实例：微信分享一个网页到朋友圈	47
3.4	常见问题	51
第 4 章	JSSDK 的图像处理	52
4.1	认识图像接口	52
4.1.1	从手机相册中选图	52
4.1.2	预览图片	55
4.1.3	上传图片	56
4.1.4	下载图片	59
4.2	实例：从手机相册中选择照片然后分享	61
4.2.1	项目需求	61
4.2.2	需求分解	61
4.2.3	代码实现	62
4.3	常见问题	65
第 5 章	JSSDK 的音频处理	66
5.1	音频接口	66
5.1.1	开始录音	66

5.1.2	停止录音.....	67
5.1.3	监听录音自动停止.....	68
5.1.4	播放语音.....	68
5.1.5	暂停播放.....	70
5.1.6	停止播放.....	71
5.1.7	监听语音播放完毕.....	71
5.1.8	上传语音.....	72
5.1.9	下载语音.....	74
5.2	常见问题.....	75
第 6 章	JSSDK 的界面操作.....	76
6.1	界面操作.....	76
6.1.1	关闭当前网页窗口.....	76
6.1.2	批量隐藏功能按钮.....	77
6.1.3	批量显示功能按钮.....	78
6.1.4	隐藏所有非基础按钮.....	80
6.1.5	显示所有功能按钮.....	80
6.2	常见问题.....	81
第 7 章	JSSDK 的地理位置.....	82
7.1	地理位置.....	82
7.1.1	获取地理位置.....	82
7.1.2	使用微信内置地图查看位置.....	85
7.2	常见问题.....	87
第 8 章	JSSDK 的智能操作与微信支付.....	88
8.1	智能接口——识别音频并返回识别结果.....	88
8.2	微信扫一扫.....	90
8.3	设备信息——获取网络状态接口.....	92
8.4	微信支付.....	93
8.4.1	微信支付公众平台申请流程.....	94
8.4.2	完成账户验证.....	95
8.4.3	设置 API 密钥.....	96

8.4.4	下载 API 证书	98
8.4.5	微信支付的开发配置	99
8.4.6	微信支付测试	99
8.5	常见问题	102
第 9 章	JSSDK 的微信商店	104
9.1	微信小店	104
9.2	微信卡券	105
9.2.1	批量添加卡券	105
9.2.2	调用适用于门店的卡券列表并获取用户选择列表	107
9.2.3	查看微信卡包中的卡券	108
9.3	常见问题	109
第 10 章	微信网页开发样式库	110
10.1	认识 WeUI	110
10.1.1	WeUI 的特色	110
10.1.2	安装 WeUI	111
10.1.3	使用 WeUI 做一个欢迎界面	113
10.2	使用表单	114
10.2.1	按钮 (Button)	114
10.2.2	表单输入 (Input)	117
10.2.3	列表 (List)	121
10.2.4	滑块 (Slider)	122
10.2.5	上传 (Uploader)	124
10.3	使用基础组件	128
10.3.1	九宫格 (Grid)	128
10.3.2	进度条 (Progress)	130
10.4	使用导航组件	133
10.5	网页其他样式	135
10.6	常见问题	136
第 11 章	实战案例 1: 中国好友挑战赛 (HTML 5 混编)	137
11.1	案例概述	137

11.2	客户需求	137
11.2.1	首页	137
11.2.2	男生游戏的逻辑	139
11.2.3	女生游戏的逻辑	142
11.2.4	分享页面	147
11.3	实现代码及分析	148
11.3.1	首页模块	148
11.3.2	参加女生组模块	164
11.3.3	参加男生组模块	198
11.4	案例总结	215
第 12 章	实战案例 2: LBS 服务获取用户位置	216
12.1	案例概述	216
12.2	实现代码及分析	216
12.3	案例总结	224
第 13 章	微信 JSSDK 相关开发技巧及问题	225
13.1	判断微信浏览器	225
13.2	微信浏览器不支持打开 App Store 页面	226
13.3	一些提示码的问题	228
13.3.1	提示码: invalid url domain	228
13.3.2	提示码: invalid signature	229
13.3.3	提示码: the permission value is offline verifying	229
13.3.4	提示码: permission denied	229
13.3.5	提示码: function not exist	229
附录 A	JS-SDK 使用权限签名算法	230
附录 B	所有 JS 接口列表	233
附录 C	所有菜单项列表	243
附录 D	卡券扩展字段及签名生成算法	245

第 1 章 认识微信公众平台



微信，英文名 WeChat，是腾讯公司在 2011 年 1 月 21 日发布的一款产品，主要是在智能终端为用户提供即时通信的免费服务。所有国人知道，微信这几年有多火，大家见面不再问电话多少，而是问：“您的微信号是什么？”让我们通过一串数字来了解一下国内的微信市场到底有多大。

截至 2016 年底，微信每月活跃用户已超过了 8 亿，汇聚的公众号超过了 1000 万，用户覆盖了 200 多个国家、超过 20 种语言，企业号的注册用户也突破了 2000 万个。微信所获取的广告收入也实现了大幅度的增长，这些广告收入主要来自于朋友圈、移动端新闻应用以及微信公众账号的效果广告。

微信不仅可以帮使用者跨平台、跨系统地进行语音通信（文字、图片、视频、语音等），还提供了很多衍生服务，例如基于位置与时间流的“摇一摇”、“漂流瓶”、“位置共享”、“公众平台”等。微信已不单单只是一个充满创新功能的手机 APP，它已成为中国移动互联网革命的代表。它覆盖了 95% 以上的智能手机，成为人们生活中不可或缺的日常使用工具，所以更多的商家进入“公众平台”领域，本书讲解的微信网页开发和 JS-SDK 便是依托于“公众平台”，旨在让商家与用户更加近距离地亲密接触。

“公众平台”是给企业、个人及一些组织性服务机构提供的一种新型开放平台。借用微信亿级的用户资源，让“公众平台账号”创造优质内容，服务于普通的大众，增加黏性，形成良性循环。“公众平台账号”简称“公众号”。

1.1 微信公众平台账号分类

目前微信“公众号”分为 4 类：订阅号、服务号、小程序和企业号。这个分类主要以

微信开放的权限为参考依据，下面分别介绍。

1.1.1 订阅号

“订阅号”的目的是为用户提供订阅信息，权限如下：

- 每天只能向订阅者（粉丝）群发一条信息。
- 发送给订阅者（粉丝）的信息，会在“订阅号”文件夹中提醒，点击进去才能进一步查看，如图 1.1 所示。



图 1.1 订阅号查看位置和打开后的效果

提醒：个人只能申请“订阅号”。

1.1.2 服务号

“服务号”的目的是为用户提供服务信息，权限如下：

- 1 个月内（自然月）仅可以群发 4 条信息。
- 发送给订阅者（粉丝）的信息，会直接显示在微信好友对话列表中，不会像订阅号

一样被折叠在一个文件夹中,如图 1.2 所示。



图 1.2 服务号查看位置

说明: 服务号申请需要提交企业或组织机构的相关信息。

服务号与订阅号在功能上的具体区别参见图 1.3。

功能权限	普通订阅号	认证订阅号	普通服务号	认证服务号
消息直接显示在好友对话列表中			✓	✓
消息显示在“订阅号”文件夹中	✓	✓		
每天可以群发1条消息	✓	✓		
每个月可以群发4条消息			✓	✓
基本的消息接收/回复接口	✓	✓	✓	✓
聊天界面底部, 自定义菜单	✓	✓	✓	✓
九大高级接口				✓
可申请开通微信支付				✓

图 1.3 服务号与订阅号在功能上的具体区别

1.1.3 企业号

企业号是“微信公众平台”的一种创新，目标是帮助企业、学校、政府机关及任何组织机构，打通横向、纵向信息流，建立上下游伙伴、内部员工之间的连接，增加协同效率，降低沟通成本，高效的运作。

微信企业号与前两个公众号最主要的区别是，只有限定范围内的用户才可以关注对应的企业号，使用企业号发送的信息数量几乎不受限制。

企业号也显示在微信的首页中，不会像订阅号一样被折叠，如图 1.4 所示。



图 1.4 企业号查看位置

1.1.4 小程序

微信在 2017 年 1 月 9 日开放了小程序，从其名字就可以看出来，这是一个支持快速注册、快速开发、快速使用、快速离开的程序。小程序最大的特色就是可以在微信内被便捷地获取和传播，同时具有出色的使用体验。

小程序在微信的“发现”栏最下角，打开后会显示曾经使用过的小程序，也可以通过

搜索栏搜索其他小程序，如图 1.5 所示。



图 1.5 小程序查看位置

1.2 微信公众号申请流程

既然微信公众号这么有用，那么作为使用者及开发者如何去申请注册呢？图 1.6 给出了一个简易的申请步骤。

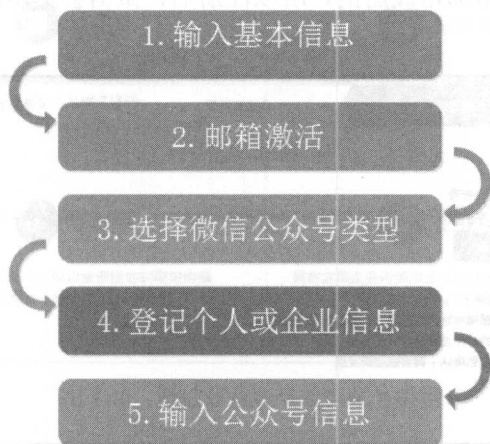


图 1.6 微信公众号申请步骤

为了方便读者操作，下面列出详细的步骤：

(1) https://mp.weixin.qq.com/cgi-bin/readtemplate?t=register/step1_tmpl&lang=zh_CN，在浏览器中输入这个链接地址，如图 1.7 所示。然后输入有效邮箱、密码及验证码，最后勾选“我同意并遵守《微信公众平台服务协议》”复选框，单击“注册”按钮。

图 1.7 微信公众账号注册页面

(2) 因为上一步是用邮箱注册的，所以注册后，微信会提示您登录邮箱进行激活，如图 1.8 所示。

图 1.8 微信公众账号邮箱激活提示

此时登录你第(1)步填写的邮箱,会收到一封激活邮件,然后单击激活链接,如图1.9所示。



图 1.9 微信公众账号邮箱激活链接

(3) 选择账号类型,如图 1.10 所示。笔者选择“订阅号”演示流程(注意,选择公众账号类型后不可更改)。



图 1.10 微信公众账号选择账号类型

(4) 选择账户类型后直接进入信息登记页面,选择主体类型,笔者选择“个人”,如图 1.11 所示。

1 基本信息 2 邮箱激活 3 选择类型 4 信息登记 5 公众号信息

用户信息登记

微信公众平台致力于打造真实、合法、有效的品牌推广平台，我们志与诚信守约、进取担当的第三方合作伙伴携手并进，建立和维护良性互动、健康有序的平台秩序。为了更好的保障你和广大微信用户的利益，请你认真填写以下登记信息。

用户信息登记审核通过后您可以：

1. 使用微信公众平台的所有功能；
2. 提高帐号可信度；
3. 点击查看微信公众平台信息登记指引。

请确认你的微信公众账号属于政府、媒体、企业、其他组织或个人，并请按照对应的类别进行信息登记。

帐号类型 订阅号

主体类型 ☐ 政府 ☐ 媒体 ☐ 企业 ☐ 其他组织 ☐ 个人

上一步 下一步

图 1.11 微信公众账号选择登记信息

选择“个人”或“企业”后，直接出现信息填写页面，目前微信主推微信支付，已经把微信支付捆绑为公众号的注册条件了，如图 1.12 所示。

主体信息登记

身份证姓名 席新亮

信息审核成功后身份证姓名不可修改；如果名字包含分隔号“/”，请勿省略。

身份证号码

请输入您的身份证号码。一个身份证号码只能注册5个公众帐号。

身份验证

为了验证你的身份，请用绑定了运营者本人银行卡的微信扫描二维码。本验证方式不扣除任何费用。

若微信没有绑定银行卡，请先绑定。如何绑定

运营者信息登记

运营者手机号码

获取验证码

请输入您的手机号码。一个手机号码只能注册5个公众帐号。

短信验证码

无法接收验证码？

请输入手机短信收到的6位验证码

图 1.12 微信公众账号填写详细登记信息

(5) 填写信息正确之后, 进入填写公众号信息页面, 填写的账号名称要合法, 不能重复, 填写完毕之后即可申请成功, 当然此时还需要微信管理员审核。如图 1.13 所示。

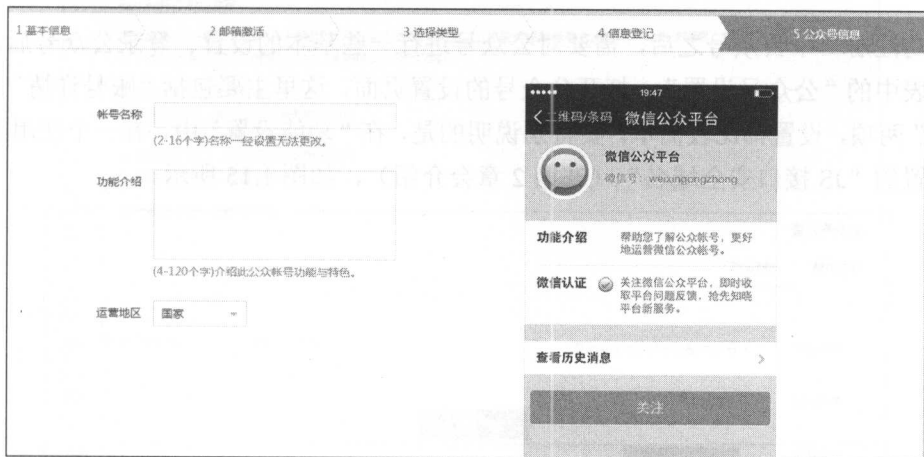


图 1.13 微信公众账号填写公众号信息

1.3 微信公众平台的认证

微信的 JSSDK 接口开发不是所有公众号都可以使用的, 这需要不同的权限, 例如自定义菜单、高级功能接口 (多客服) 这些接口就需要被认证过的公众号才可以使用。

目前“微信认证”暂时不支持主体类型是“个人”类型的, 如果读者朋友有企业的账号可以申请认证。登录微信公众号后, 单击左侧列表中的“微信认证”就可以认证公众号了。如果微信认证通过, 会显示如图 1.14 所示的界面。

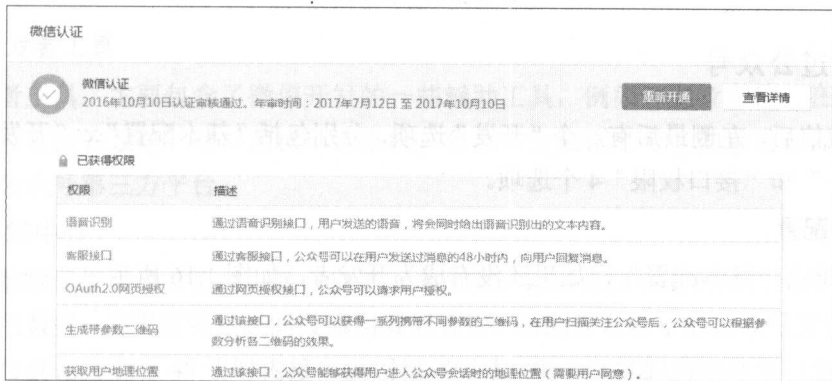


图 1.14 微信认证成功之后界面

1.4 公众号设置

成功注册一个公众号之后，需要对公众号进行一些基本的设置。登录公众号后，单击左侧列表中的“公众号设置”，打开公众号的设置页面，这里主要包括“账号详情”与“功能设置”两项，设置都比较简单。要特别说明的是，在“功能设置”中，有一个使用 JSSDK 需要设置的“JS 接口安全域名”（在第 2 章会介绍），如图 1.15 所示。

公众号设置		
帐号详情		功能设置
隐私设置	已允许 通过名称搜索到本帐号	设置
图片水印	不添加水印	设置
业务域名		设置
JS接口安全域名		设置
网页授权域名		设置

图 1.15 公众号设置的功能设置界面

1.5 成为微信开发者

成为微信开发者有两种方式：一种是在任意公众号内部，通过开通成为开发者；一种是无须公众账号、快速申请接口测试号，只要微信登录就可以。

1.5.1 通过公众号

登录微信后，左侧最后有一个“开发”选项，分别包括“基本配置”、“开发者工具”、“运维中心”和“接口权限”4 个选项。

1. 基本配置

(1) 单击“基本配置”，如果还没有成为开发者，如图 1.16 所示。

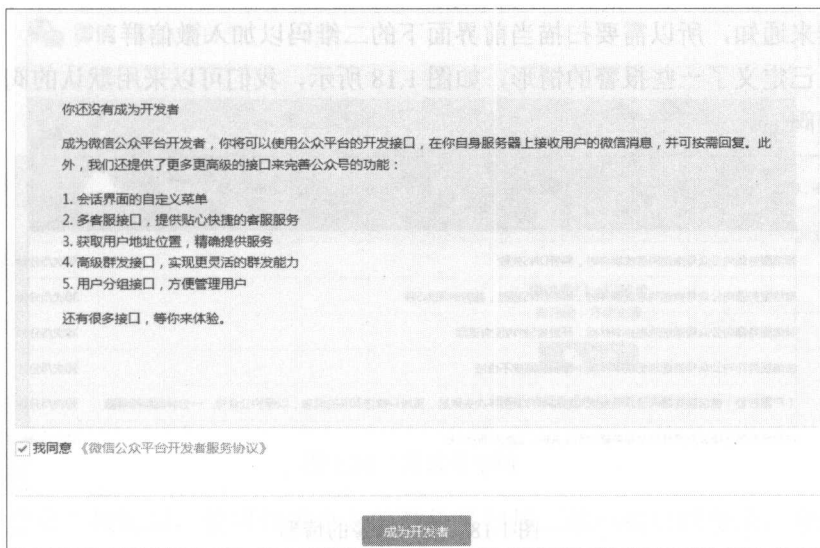


图 1.16 成为开发者

(2) 勾选“我同意”复选框，然后单击“成为开发者”按钮，就会出现我们需要的开发者 ID，如图 1.17 所示。



图 1.17 开发者 ID

2. 开发者工具

开发者工具，主要包含了微信开发的一些辅助工具，例如开发者文档、在线接口调试工具、开发者问答系统、接口测试申请系统、公众平台测试账号（如果没有开发权限可以试用）、公众号第三方平台。

3. 运维中心

包括数据监控、日志查询、接口报警三个选项。数据监控功能实时统计公众号接口/被动回复调用数据，并支持特定时间段调用数据查询，帮助开发者了解公众号接口调用情况。日志查询可查询公众号 48 小时内接口调用错误日志。接口报警的功能是专门为开发者设计的，它是为了帮助开发者在自身服务器和接口响应出现问题时，尽快发现问题。接口报警

通过微信群来通知，所以需要扫描当前界面下的二维码以加入微信群。

微信自己定义了一些报警的情形，如图 1.18 所示，我们可以采用默认的阈值，也可以设计自己的阈值。

报警规则			
发生下述情形，即超过报警阈值时，将进行报警（若已开通微信支付权限，则在微信支付消息通知时，也根据下述情形报警）：			
情形	描述	报警阈值	操作
DNS失败	微信服务器向公众号推送消息或事件时，解析DNS失败	30次/5分钟	设置阈值
DNS超时	微信服务器向公众号推送消息或事件时，解析DNS超时，超时时间为5秒	30次/5分钟	设置阈值
请求超时	微信服务器向公众号推送消息或事件后，开发者5秒内没有返回	30次/5分钟	设置阈值
回应失败	微信服务器向公众号推送消息或事件后，得到的回应不合法	30次/5分钟	设置阈值
MarkFail	（严重警告）微信服务器向公众号推送消息或事件发生多次失败后，暂时只推送30%的消息，以保护公众号。一分钟后解除屏蔽	30次/5分钟	设置阈值
连接超时	微信服务器连接公众号开发者服务器时发生超时，超时时间为5秒	30次/5分钟	设置阈值

图 1.18 接口报警的情形

4. 接口权限

显示接口权限列表，主要告诉当前“公众号”的拥有者有哪些权限能开发，如图 1.19 所示。

接口权限表					
类目	功能	接口	每日调用上限/次	接口状态	操作
对话服务	基础支持	获取access_token	2000	已获得	
		获取微信服务器IP地址		已获得	
	接收消息	验证消息真实性	无上限	已获得	
		接收普通消息	无上限	已获得	
		接收事件推送	无上限	已获得	
		接收语音识别结果（已开启）	无上限	已获得	关闭
	发送消息	自动回复	无上限	已获得	
		客服接口	500000	已获得	
		群发接口	详情 >	已获得	
		模板消息（业务通知）		未获得	

图 1.19 部分“接口权限列表”

1.5.2 通过测试号

微信为开发者提供了测试号，无须开通公众号就可以申请接口测试，直接体验和测试公众平台所有高级接口。输入网址 <https://mp.weixin.qq.com/debug/cgi-bin/sandbox?t=sandbox/login>，就可以打开申请界面，如图 1.20 所示。

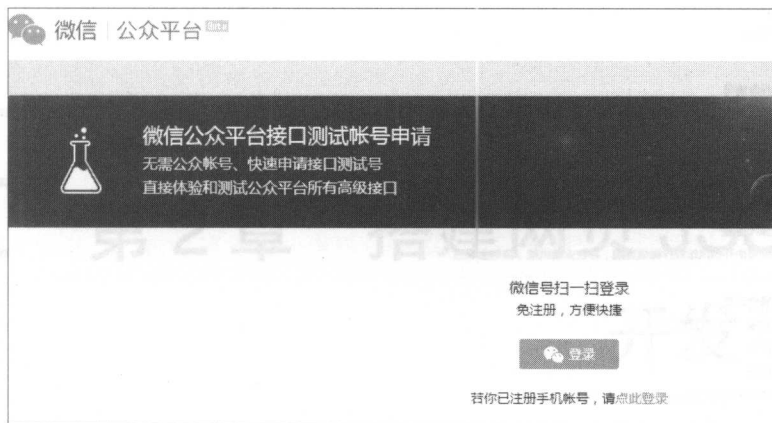


图 1.20 测试号申请

单击“登录”按钮后，使用任意个人的微信号扫描二维码就可以登录，登录后就会获得一个开发者 ID，如图 1.21 所示。

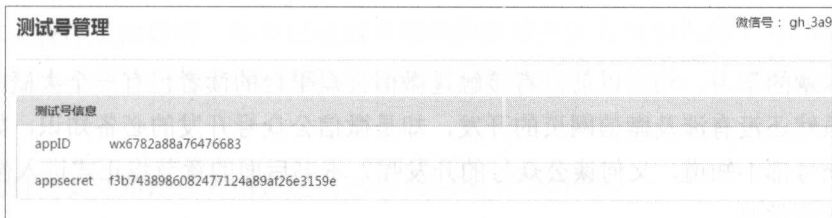


图 1.21 开发者 ID

在测试时如果没有自己的服务器，可以使用一些免费的云服务器来解决，目前国内使用最多的是新浪的 SAE，注册送 200 个云豆，可以免费使用一周左右。

1.6 公众号的安全中心

登录微信后在左侧栏中有个“安全中心”选项，腾讯的安全措施做得不错，这个模块中包含了几大安全控制的功能，如图 1.22 所示。

在“安全中心”里，我们可以修改微信登录密码、可以设置管理员微信号（发布微信时需要用管理员微信的“扫一扫”功能才能发布）、可以设置运营者微信号、可以对“群发”功能实现风险保护，还可以对“开发者中心”重置 AppSecret 的保护。

作为开发人员或一个平台的维护人员，安全中心是我们在运营产品前必须了解的一个环节。

安全中心		
管理员微信号	你的帐号已绑定管理员微信号及1个长期运营者，0个短期运营者微信号	详情
风险操作保护	你的帐号已开启风险操作微信保护，进行风险操作时需进行微信号验证	详情
风险操作提醒	你的帐号未开启风险操作微信提醒，存在安全风险，请马上设置	详情
风险操作记录	最近操作： 暂无操作记录	详情
修改密码	用管理员微信扫码验证后，修改你的密码	修改

图 1.22 安全中心功能列表

1.7 本章小结

通过本章的学习，相信以前没有接触过微信公众平台的读者也有一个大概的全貌认知了。本章虽然还没有涉及微信网页的开发，却是微信公众号开发的必备知识，如果我们连什么是公众号都不知道，又何谈公众号的开发呢？本书后面的章节将正式进入微信 JSSDK 和网页开发的学习。

第2章 搭建网页 JSSDK 开发环境



2015 年伊始, 微信便开放了 JSSDK, 从此 JSSDK 以惊人的速度打造了庞大的 HTML 5 微信链条。很多寄生于微信平台下的第三方企业热捧微信的公众号开发体系, 同时这一 SDK 也受到了广大程序员的好评。如今已经越来越多的微信开发人员加入到了 HTML 5 开发的行列中, 所以本书更详细地介绍 JSSDK+HTML5 的微信网页应用, 让读者能更好地学习 JSSDK。

本章的目的就是帮助读者搭建起自己的 JSSDK 开发环境。

2.1 JSSDK 介绍

在学习 JSSDK 之前, 首先要知道 JSSDK 到底是什么, 到底符不符合自己的项目情况。下面笔者就基于官方说明及个人理解, 梳理一下。

1. 面向的对象

JSSDK 主要是针对在微信内部做网页开发的同学们发放的福利, 在 JSSDK 的接口范围内想怎么玩就怎么玩, 同时 JSSDK 也支持网页内微信支付, 便利了微信商城的开发。

2. 截止到 2017 年 2 月 1 日开放的接口

主要包括基础接口、分享接口、图像接口、音频接口、智能接口、设备信息、地理位置、摇一摇周边、界面操作、微信扫一扫、微信小店、微信卡券、微信支付 (具体的接口列表在 2.5 节中介绍)。

3. 接口统一说明

JSSDK 的所有接口都通过“wx”（或 jWeixin）对象调用，参数以对象形式配置。除了每个接口特有的参数外，所有的接口还有如下一部分通用参数：

- success，调用成功时执行的回调函数。
- fail，调用失败时执行的回调函数。
- complete，无论成功与失败，只要调用完成，就会执行的回调函数。
- cancel，用户取消某项操作时的回调函数。
- trigger，监听 Menu 中按钮交互时的方法，此参数仅支持 Menu 中的相关 API。

4. JSSDK 的大致部署过程

依据账号类型不同，部署有如下两种形式：

1) 普通公众账号，大多数开发者都是基于此做开发的。

(1) 绑定域名。

(2) 将 JS 文件引入。

(3) 配置 config 接口权限，主要目标是获取 access_token 与 jsapi_ticket，再通过 signature 进行配置，这几个关键的“配置点”是以后操作 JSSDK 的重要依据。

(4) 验证是否接入成功，其中有两个回调函数用于确定结果（ready 表示接入成功，error 表示接入失败）。

2) 第三方平台开发者，这类开发与第一种差别不大，只是在接入方式上略有差异。

(1) 填写网页的开发域名，将此域名作为旗下的授权公众号，简单的理解就是申请第三方平台开发权限（详情请参考官方的接入前必读-申请资料说明）。

(2) 与第一种一样，引入 JS 文件。

(3) 配置 config 接口验证，与第一种唯一的区别是在获取 jsapi_ticket 时，不是通过公众号的 access_token 获取，而是通过第三方平台的授权公众号 token 获取，另外，在 JSSDK 的其他配置中，其他信息都是正常的公众号资料，而非第三方的。

(4) 验证是否接入成功，其中有两个回调函数用于确定结果（ready 表示接入成功，error 表示接入失败）。

特别说明：2.2 节的代码适用于有自己的服务器并且有自己认证的公众号的读者；2.3 节的代码适合使用微信测试号并且公众号还没有认证，但是想学习微信网页开发的读者。

2.2 部署 JSSDK

部署 JSSDK 并不难，有时候需要一点后台知识，但也不是太难的那种。本节主要用 PHP 作为后台参考语言，为了照顾初学者，会把代码写得简单一点。

在 2.1 节中已经讲解了部署的大致流程，现在与笔者一起详细地部署一下吧。

2.2.1 绑定域名

如果读者是第 1 次绑定域名，则可按照下面的步骤操作：

(1) 首先需要登录微信公众平台。

(2) 选择左侧列表栏的“公众号设置”选项，如图 2.1 所示。

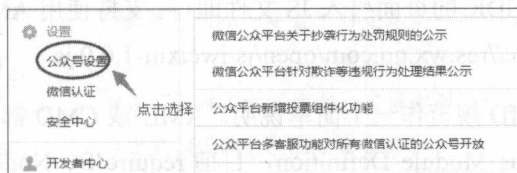


图 2.1 选择“公众号设置”选项

(3) 在打开的“公众号设置”界面中，单击“功能设置”选项卡，然后填写 JS 接口的安全域名。填写的域名一定要合法并且备案，如图 2.2 和图 2.3 所示。

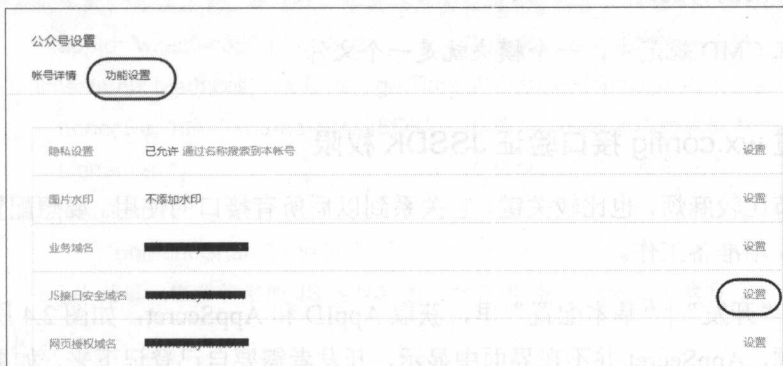


图 2.2 单击“功能设置”选项卡 JS 接口安全域名

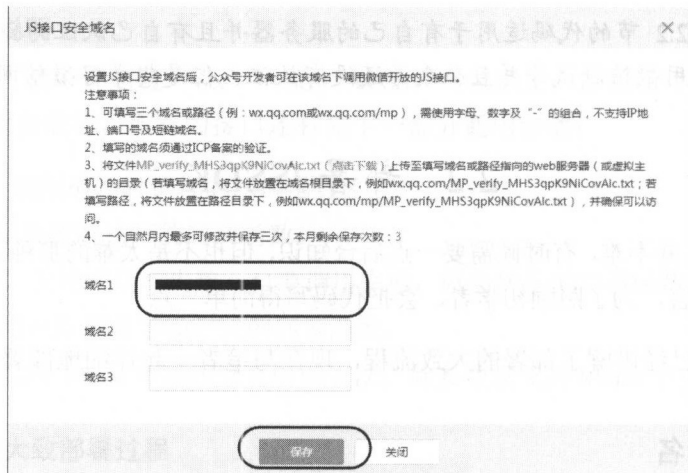


图 2.3 填写 JS 接口安全域名

2.2.2 引入 JS 文件

在需要调用微信 JSSDK 的页面引入 JS 文件即可，支持使用 AMD 或 CMD 的方式加载文件，文件地址为：<http://res.wx.qq.com/open/js/jweixin-1.0.0.js>。

这里对 AMD 和 CMD 规范作一个简单说明。AMD 或 CMD 都是模块加载规范，AMD 规范全称是 Asynchronous Module Definition，目前 requireJS、Node.js、Dojo、jQuery 都支持 AMD 规范。CMD 规范的全称是 Common Module Definition，该规范明确了模块的基本书写格式和基本交互规则。不管是 AMD 还是 CMD，这些规范的目的都是为了实现 JavaScript 的模块化开发，特别是在浏览器端的开发。目前这些规范的实现都能达成在浏览器端模块化开发的目的。

说明：在 CMD 规范中，一个模块就是一个文件。

2.2.3 通过 wx.config 接口验证 JSSDK 权限

这个环节比较麻烦，也比较关键，它关系到以后所有接口的使用。要想配置 wx.config，前期需要做 5 项准备工作。

(1) 在“开发”|“基本配置”里，获取 AppID 和 AppSecret，如图 2.4 所示。目前考虑到安全问题，AppSecret 并不在界面中显示，开发者需要自己登记下来，如果忘记，则需要重置。

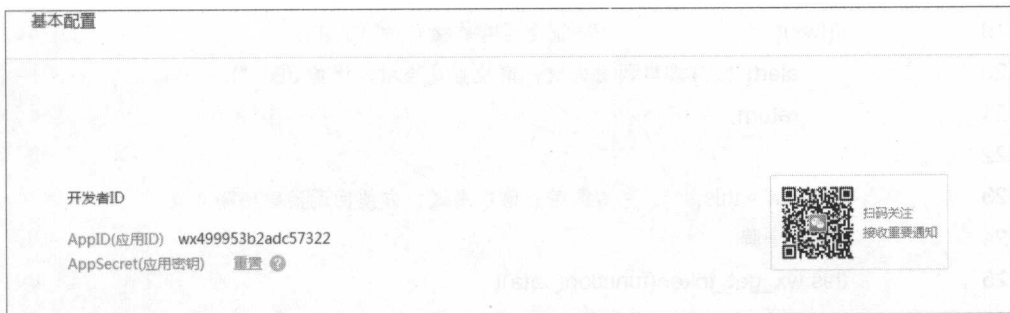


图 2.4 微信后台→开发者 ID

在开始介绍下一步之前，首先搭建一个通用的微信 JSSDK 库，以方便流程讲解，并且可以作为将来项目的公共模块使用。此库依赖 index.html、PHP 脚本、外部库 3 个文件。

注意：下面的代码比较多，如果读者想尽快阅读，可以直接从本书源码中获取以下文件。如果要了解代码流程，则可参考代码中的注释。

下面是“wxJSSDK”库的 JS 代码：

```

01  var wxJSSDK = {           //声明微信全局变量，防止污染外部环境
02      version:"1.0",        //版本号
03      appName:"",           //使用当前库的开发者，可以配置应用的名字
04      isReady:false,        //微信 JSSDK 是否初始化完毕
05      access_token:"",      //令牌
06      ticket:"",            //微信临时票据
07      config:{
08          debug: true, // 开启调试模式，调用的所有 API 的返回值会在客户端 alert 出来，若要查
09              看传入的参数，可以在 PC 端打开，参数信息会通过 log 打出，仅在 PC 端时才会打印
10          appId: 'wxecfac3508230c6c0', // 必填，公众号的唯一标识
11          timestamp: Math.ceil(new Date().getTime()/1000).toString(), // 必填，生成签名的时间戳
12          nonceStr: 'html5waibao_wxJSSDK', // 必填，生成签名的随机串
13          signature: "", // 必填，签名，见附录 A
14          jsApiList: [
15              "onMenuShareTimeline"
16          ] // 必填，需要使用的 JS 接口列表，所有 JS 接口列表见附录 B
17      },
18      init:function(){       //函数功能：初始化

```



```

19         if(!wx){                                //验证是否存在微信的 JS 组件
20             alert("微信接口调用失败，请检查是否引入微信 JS! ");
21             return;
22         }
23         var that = this;                          //保存当前作用域，方便回调函数使用
24         //获取令牌
25         this.wx_get_token(function(data){
26             if(data.access_token){
27                 Cookie.Set("access_token", data.access_token, 3600);
28                 that.access_token = data.access_token;
29             }
30             //获取票据
31             that.wx_get_ticket(function(data){
32                 if(data.ticket){
33                     Cookie.Set("ticket", data.ticket, 3600);
34                     that.ticket = data.ticket
35                 }
36                 //获取签名
37                 that.wx_get_signature(function(data){
38                     that.config.signature = data;
39                     that.initWx(function(){          //初始化微信接口
40                         //初始化完成后的执行
41                     });
42                 });
43             });
44         });
45     },
46     //获取令牌
47     wx_get_token:function(call){
48         this.access_token = Cookie.Get("access_token");
49         if(!Cookie.Get("access_token")){
50             $.get("./API/wx_get_token.php", {},
51                 function(data){
52                     debugger

```



```

53             call && call(data);
54         }, "json");
55     return;
56 }
57     call && call({});
58 },
59 //获取票据
60 wx_get_ticket:function(call){
61     this.ticket = Cookie.Get("ticket");
62     if(!this.ticket){
63         $.get("./API/wx_get_jsapi_ticket.php", {access_token: this.access_token},
64             function(data){
65                 debugger
66                 call && call(data);
67             }, "json");
68     return;
69     }
70     call && call({});
71 },
72 //获取签名
73 wx_get_signature:function(call){
74     $.get("./API/wx_get_signature.php", {
75         ticket: this.ticket,           // 必填，生成签名的时间戳
76         timestamp: this.config.timestamp, // 必填，生成签名的时间戳
77         nonceStr: this.config.nonceStr, // 必填，生成签名的随机串
78         url: window.location.href      // 必填，生成签名的随机串
79     },
80     function(data){
81         debugger
82         call && call(data);
83     });
84 },
85 initWx:function(call, errorCall){//初始化微信接口
86     var that = this;

```

```

87      wx.config(this.config);//初始化配置
88      /*config 信息验证后会执行 ready 方法，所有接口调用都必须在 config 接口获得结果之
89      *后。config 是一个客户端的异步操作，如果需要在页面加载时就调用相关接口，
90      *则必须把相关接口放在 ready 函数中调用来确保正确执行。对于用户触发时才调用的
91      *接口，则可以直接调用，不需要放在 ready 函数中。
92      */
93      wx.ready(function(){
94          that.isReady = true;
95          console.log("初始化成功")
96          call && call();
97      });
98      /*config 信息验证失败会执行 error 函数，如签名过期导致验证失败。
99      *具体错误信息可以打开 config 的 debug 模式查看，也可以在返回的 res 参数中查看，
100      * 对于 SPA 可以在这里更新签名。
101      */
102      wx.error(function(res){
103          that.isReady = "false";
104          errorCall && errorCall();
105      });
106  }
107 }
108 //执行初始化
109 wx.JSSDK.init();

```

要想运行以上代码，还要写一个 JS 环境，如 index.html，代码如下：

```

01  <!DOCTYPE html>
02  <html lang="en">
03  <head>
04      <meta charset="UTF-8">
05      <title>第 1 章 1.1 节</title>
06      <script src="js/jquery-1.11.2.min.js"></script>
07      <script src="http://res.wx.qq.com/open/js/jweixin-1.0.0.js"></script>
08      <script src="js/cookie.js"></script>
09      <script src="js/wxJSSDK.js"></script>

```

```

10 </head>
11 <body>
12     <h1 style="font-size: 40px">.</h1>
13     <b style="font-size: 20px">JSSDK 部署学习! </b>
14 </body>
15 </html>

```

以上代码的依赖库如下:

- jQuery 库 1.11.2 版本 (请在官网下载, 链接为 <http://jquery.com/download/>);
- 微信的 JSSDK 链接 <http://res.wx.qq.com/open/js/jweixin-1.0.0.js>;
- cookie.js 在源码中存在, 用于前端的 Cookie 操作。

“wxJSSDK”还依赖一些 PHP 脚本, 暂时只是贴出代码, 仅供读者临时参阅。如果看不懂代码也没有关系, 随着后面接口的应用调试, 读者会逐步明白。

说明: 后台采用的是 PHP, 如果您是 PHP 小白, 关于 PHP 环境的搭建, 推荐使用一键安装的 wampserver。

获取“令牌”(access_token)的 PHP 文件 wx_get_token.php, 其代码如下:

```

01 <?php
02 $token = fopen("./cache/access_token.txt","r");//打开 access_token 文件流
03 $token = fread($token,1023);//获取 token
04 $tokenTime = fopen("./cache/access_tokenTime.txt","r");//打开 access_tokenTime 文件流
05 $prevTime = intval(fread($tokenTime,1023));//获取上次缓存文件的时间
06 $nowTime = time()-$prevTime;//计算时间差值
07 if ($nowTime>= 3600) { //如果时间差值大于等于 3600 秒, 那么就重新获取 access_token
08     $res =
09     file_get_contents('https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential
10     &appid=wx60cc0b8b6bb0e83b&secret=a2f09aa037f335194e2dafb5d157a55f');
11     $res = json_decode($res, true);
12     $token = $res['access_token'];
13     //$json_string = json_encode($res,JSON_UNESCAPED_UNICODE);
14     // 注意: 这里需要将获取到的 token 缓存起来 (或写到数据库中)
15     // 不能频繁地访问 https://api.weixin.qq.com/cgi-bin/token, 因为每日有次数限制
16     // 通过此接口返回的 token 的有效期限目前为 2 小时。令牌失效后, JSSDK 也就不能用了。

```



```

17      // 因此，这里将 token 值缓存 1 小时，比 2 小时小。
18      // 缓存失效后，再从接口获取新的 token，
19      // 这样就可以避免 token 失效
20
21      //缓存 token 码
22      $myfile = fopen("./cache/access_token.txt", "w") or die("Unable to open file!");
23      fwrite($myfile, $token);
24      fclose($myfile);
25      //最后缓存时间
26      $myfileTime = fopen("./cache/access_tokenTime.txt", "w") or die("Unable to open file!");
27      fwrite($myfileTime, time());
28      fclose($myfileTime);
29      echo json_encode($res);
30  }else{
31      echo '{"access_token":"'.$token.'"}';
32  }

```

获取“临时签名”（ticket）的 PHP 文件 wx_get_jsapi_ticket.php，其代码如下：

```

01  <?php
02      $ticket = fopen("./cache/ticket.txt", "r");//打开 ticket 文件流
03      $ticket = fread($ticket, 1023);//获取 ticket
04      $ticketTime = fopen("./cache/ticketTime.txt", "r");//打开 ticketTime 文件流
05      $prevTime = intval(fread($ticketTime, 1023));//获取上次缓存文件时间
06      $nowTime = time() - $prevTime;//计算时间差值
07      if ($nowTime >= 3600) { //如果时间差值大于等于 3600 秒，那么就重新获取 ticket
08          do{
09              $token = $_GET['access_token'];
10              $url2 =
11          sprintf("https://api.weixin.qq.com/cgi-bin/ticket/getticket?access_token=%s&type=jsapi",
12              $token);
13              $res = file_get_contents($url2);
14              $res = json_decode($res, true);
15              $ticket = $res['ticket'];
16          }while(0);
17

```

```

18          // 注意：这里需要将获取到的 ticket 缓存起来（或写到数据库中）
19          // ticket 和 token 一样，不能通过频繁地访问接口来获取，因而在每次获取后，
20          // 需要把它保存起来。
21          // 缓存临时票据
22          $myfile_ = fopen("./cache/ticket.txt", "w") or die("Unable to open file!");
23          fwrite($myfile_, $ticket);
24          fclose($myfile_);
25          //缓存时间
26          $myfile_Time = fopen("./cache/ticketTime.txt", "w") or die("Unable to open file!");
27          fwrite($myfile_Time, time());
28          fclose($myfile_Time);
29
30          echo json_encode($res);
31      }else{
32          echo '{"ticket":"","ticketTime":""}';
33      }

```

获取“签名”（signature）的 PHP 文件 wx_get_signature.php，其代码如下：

```

01  <?php
02      $timestamp = $_GET['timestamp'];
03      $wxnonceStr = $_GET['nonceStr'];
04      $wxticket = $_GET['ticket'];
05      //将 ticket、noncestr、timestamp、分享的 url 按字母顺序连接起来，进行 sha1 签名
06      $wxOri = sprintf("jsapi_ticket=%s&noncestr=%s&timestamp=%s&url=%s",
07          $wxticket, $wxnonceStr, $timestamp,
08          $_GET['url']
09      );
10      $wxSha1 = sha1($wxOri);
11      echo $wxSha1;

```

最后不要忘记，PHP 脚本需要 cache 文件夹、access_token.txt（空文件）、access_tokenTime.txt（文件内容是 0）、ticket.txt（空文件）以及 ticketTime.txt（文件内容是 0）做缓存，全部代码结构如图 2.5 所示。

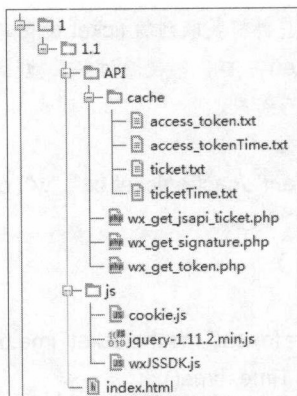


图 2.5 wxJSSDK 项目代码结构

看完上面的所有代码后，读者应该有个基本认知了。由于代码注释比较详细，所以笔者只是帮助读者再梳理一下大致逻辑，请结合图 2.6 查看。

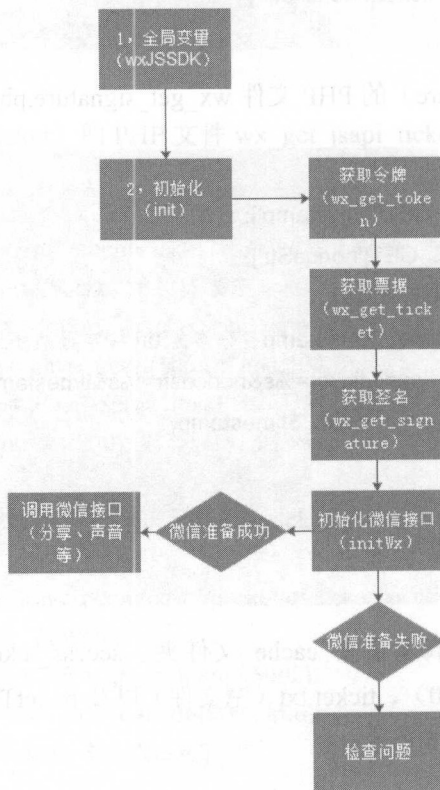


图 2.6 wxJSSDK 逻辑思路示意图

第1步, 创建全局变量“wxJSSDK”, 将所有与 JSSDK 有关的模块放入其中, 防止污染外部环境。

第2步, 做初始化, 包括获取令牌、获取票据、获取签名。

第3步, 将之前获取的数据信息, 配置 wx.config 用以初始化 wx。

第4步, 如果 wx 的初始化状态成功, 就可以使用微信 JSSDK 了。

(2) 获取令牌, 微信的 JSSDK 会限制令牌的使用期限, 限制调用次数与频率, 有效期是 7200 秒。笔者缓存 1 小时, 代码中采用的是 PHP 文件缓存与前端 Cookie 缓存, 防止多次请求。“wxJSSDK.wx_get_token”通过“cookie”判断是否请求脚本“wx_get_token.php”。如果“cookie”存在令牌, 则直接调用回调函数, 否则就发起请求。在 PHP 脚本中, 首先读取缓存文件“access_token.txt”与“access_tokenTime.txt”中的内容, 然后判断时间差值是否“>=”3600 秒(1 小时), 如果是, 则用“file_get_contents”获取令牌, 并且进行文件缓存, 并将结果返回给前端的 JS, JS 再进行“cookie”缓存, 之后再调用回调函数, 如图 2.7 所示。

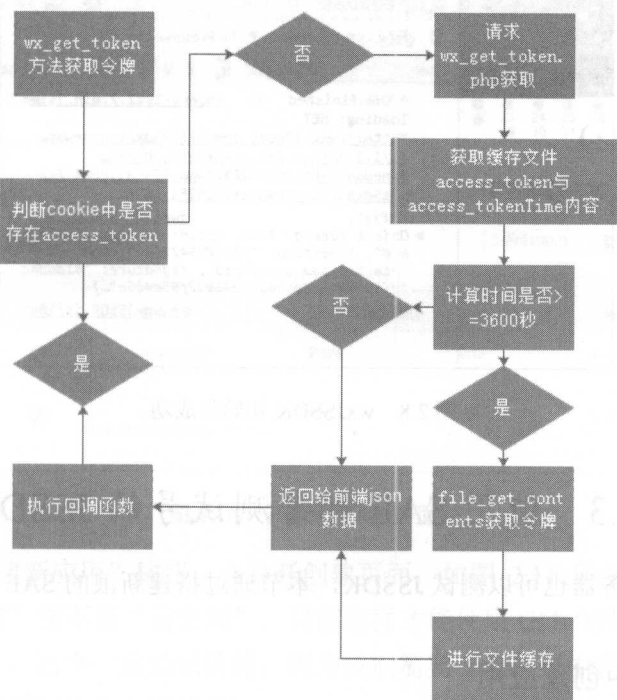


图 2.7 获取令牌逻辑图

(3) 通过“access_token”获取 jsapi 的 ticket，微信“wxJSSDK”中的“wx_get_ticket”方法用来判断“cookie”是否存在，如果不存在，就请求“wx_get_jsapi_ticket.php”接口获取 ticket。这次需要传递参数“access_token”，“wx_get_jsapi_ticket.php”中的数据也是通过文件缓存与“file_get_contents”获取“ticket”，将获取的数据响应给“wx_get_ticket”，流程思路与图 2.7 类似。

(4) 获取签名，主要的算法是将 ticket、noncestr、timestamp 以及当前的 url（不包含#及其后面部分）按字母顺序连接起来，进行 SHA1 签名。另外，如果想验证自己的签名是否符合 JSSDK 的相关规则或是否生效，可以访问微信的 JS 接口签名校验工具，链接为：<http://mp.weixin.qq.com/debug/cgi-bin/sandbox?t=jsapisign>。

(5) 利用以上获取到的数据，配置 wx.config。如果成功，“wx.ready”会运行；如果失败，“wx.error”会给出提示。

最后，运行一下本节的代码吧，如果 Chrome 浏览器出现如图 2.8 所示的效果，控制台打印“初始化成功”，说明您的代码成功了。

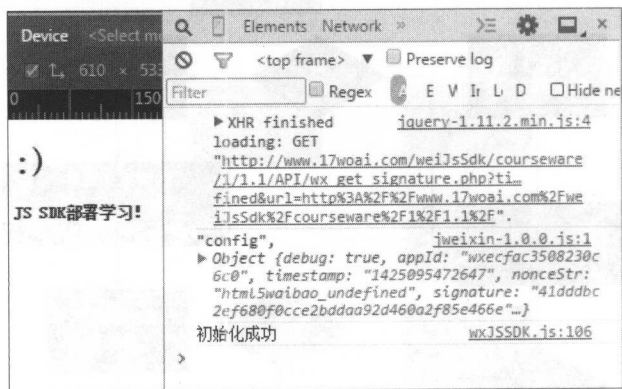


图 2.8 wxJSSDK 初始化成功

2.3 使用 SAE 部署测试号的 JSSDK

没有自己的服务器也可以测试 JSSDK，本节通过搭建新浪的 SAE 来实现。

2.3.1 在 SAE 中创建应用

打开网址 <http://www.sinacloud.com/>，单击右上角的“注册账号”按钮，如果有微博，

可以使用已有的微博账号注册，注册过程不再赘述。每次登录需要用微博账号扫描。

(1) 登录后，在右上角的控制台打开“云应用 SAE”，如图 2.9 所示。

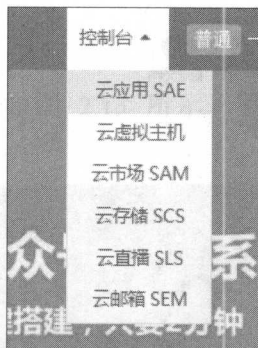


图 2.9 云应用 SAE

(2) 打开后看到如图 2.10 所示页面，这里笔者已经新建了两个应用，初次创建时，下面的应用管理显示为空。

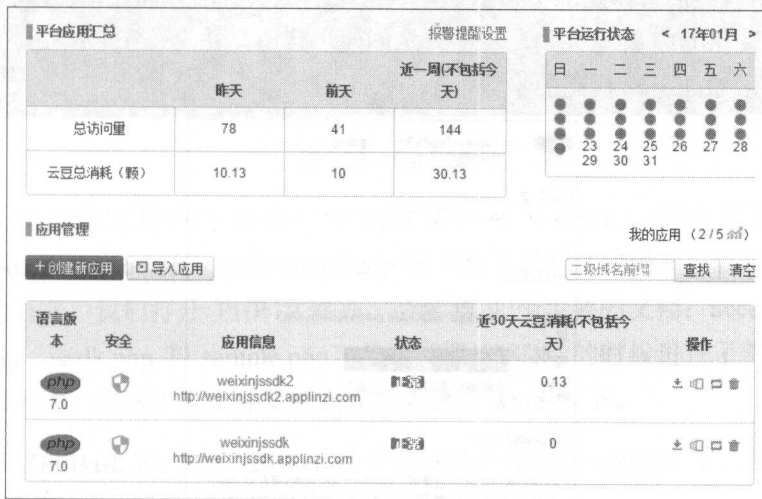


图 2.10 云应用 SAE 界面

(3) 单击“创建新应用”按钮，会打开创建页面，如图 2.11 所示。注意，这里我们选择的是“标准环境”而不是“云空间”，只有这样才能使用 GIT 代码进行管理。下面会有一个“二级域名”，这个一定要记清楚，因为我们需要在测试号中设置“JS 接口安全域名”，那时候就需要填写这个二级域名。

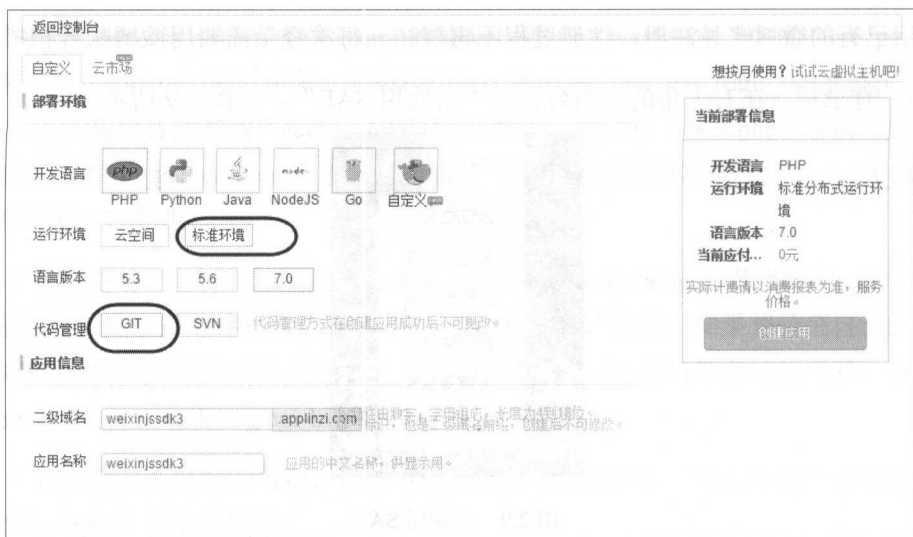


图 2.11 创建新应用

(4) 单击右侧的“创建应用”按钮，就会进入代码管理界面，如图 2.12 所示。因为我们选择的是使用 GIT 管理代码，目前只是创建了应用，并没有在应用中添加代码，所以图中会给出使用 GIT 管理代码的步骤。



图 2.12 代码管理

(5) 不会使用 GIT 的读者，可以查看一些 GIT 的资料。我们在这里也手把手地指导读者进行配置。首先下载一个 GIT 客户端 Git-2.8.1-64-bit，读者可以从网盘或搜索引擎下

载。下载安装后会会有一个 Git Bash 选项，打开它就是我们需要的操作界面。这是传统 DOS 界面，通过命令来实现代码的管理。

2.3.2 上传测试代码

(1) 打开 Git Bash，效果如图 2.13 所示。默认操作文件夹是在当前电脑的 C:\Users\Administrator 下。

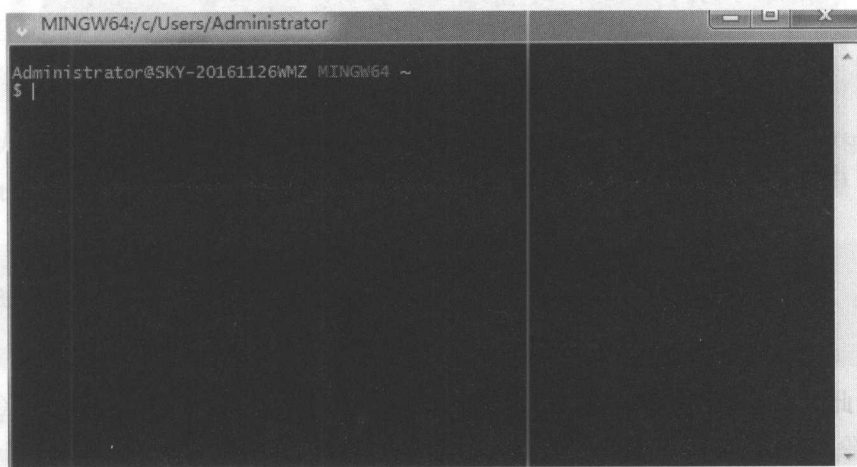


图 2.13 Git Bash

(2) 我们在此文件夹下创建一个测试文件夹 wxtest6，然后到微信官方文档 <http://demo.open.weixin.qq.com/jssdk/sample.zip> 最下面下载测试代码。下载后的文件夹包含了所有支持的语言。我们打开 PHP 文件夹，会看到 4 个主要的文档：access_token.php、jsapi_ticket.php、jssdk.php 和 sample.php。前面部署 JSSDK 的时候进行了整理，这里我们可以直接拿来测试。将这 4 个文件复制到我们新建的 wxtest6 下。

(3) 打开 Git Bash 客户端，使用 cd 命令将文件夹切换到 wxtest6 下，然后使用以下命令，操作结果如图 2.14 所示。

```
$ git init      #初始化
$ git remote add sae https://git.sinacloud.com/weixinjssdk3  #远程添加，最后是自己的应用名
$ git add --all      #变动所有文件
$ git commit -m "initial commit"  #提交，最后“”内的内容是备注，说明本次操作
$ git push sae master:1  #将内容推送到主分支
```

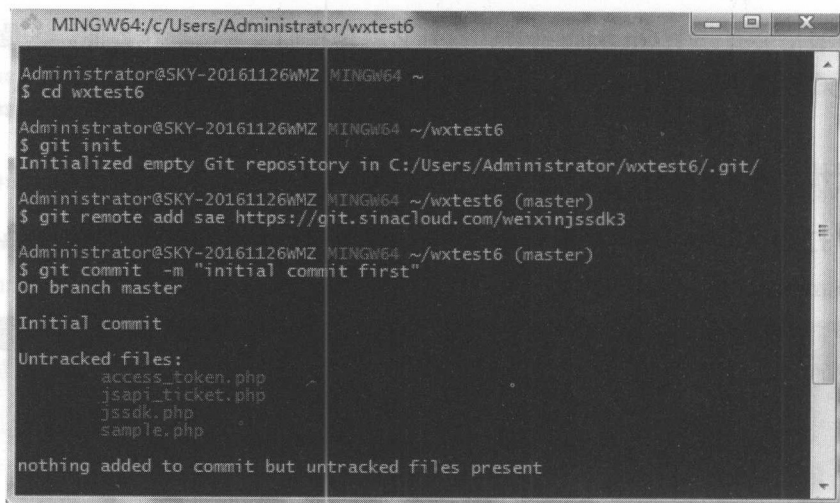


图 2.14 Git Bash 操作步骤

提示：执行最后一步时要输入的用户名和密码是应用创建时的账号，不是微博登录的账号。

(4) 此时在任意浏览器中输入 <http://1.weixinjssdk3.applinzi.com/sample.php>（因为测试过程中删除 SAE 的应用后不能新建同名应用，所以笔者显示的是新应用 `weixinjssdk6`），如果不输出任何内容，就说明我们的服务器配置成功了，如图 2.15 所示。`sample.php` 里面因为 body 里什么也没有，所以显示为空。后面正式使用 JSSDK 接口时，要填写的内容就会在 `sample.php` 中显示。

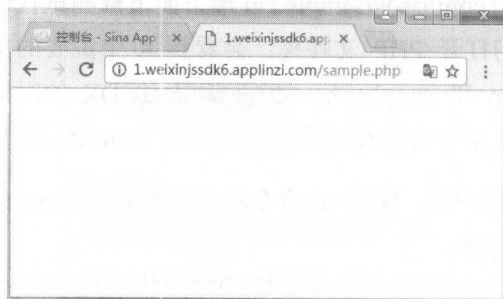


图 2.15 sample.php 的输出内容

2.3.3 绑定 JS 接口

测试号需要在测试号管理界面绑定 JS 接口，如图 2.16 所示。这个域名是上面配置好

的二级域名。单击“修改”按钮可以更新域名。

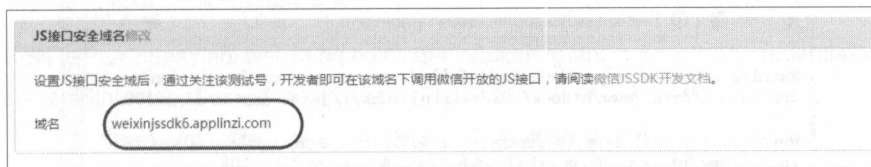


图 2.16 绑定 JS 接口

此时我们并没有使用测试号，打开项目中的 `sample.php` 文件，将如下第 3 行代码：

```
$jssdk = new JSSDK("yourAppID", "yourAppSecret");
```

更换为自己的 AppID 和 AppSecret，打开测试号管理界面就能看到这两项内容。然后再打开 Git Bash 更新远程项目。使用如下代码：

```
$ git add --all
$git commit -m "update sample"
$git push sae master:1
```

效果如图 2.17 所示。

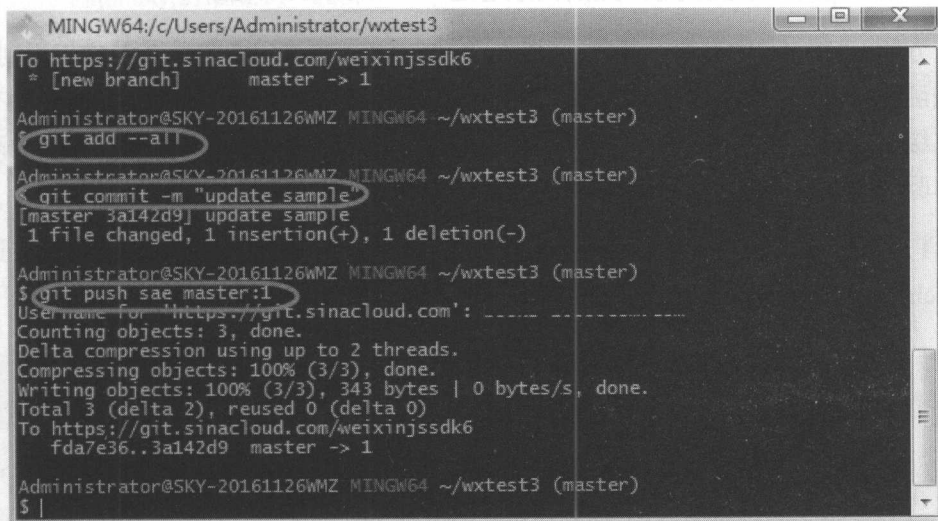


图 2.17 更新远程项目

注意，此时使用 SAE 测试项目会出现问题，因为微信操作中会有 `fopen`、`fwrite` 等 I/O 操作，而 SAE 并不支持此操作，所以此时在浏览器中测试应用，会有如图 2.18 的提示。

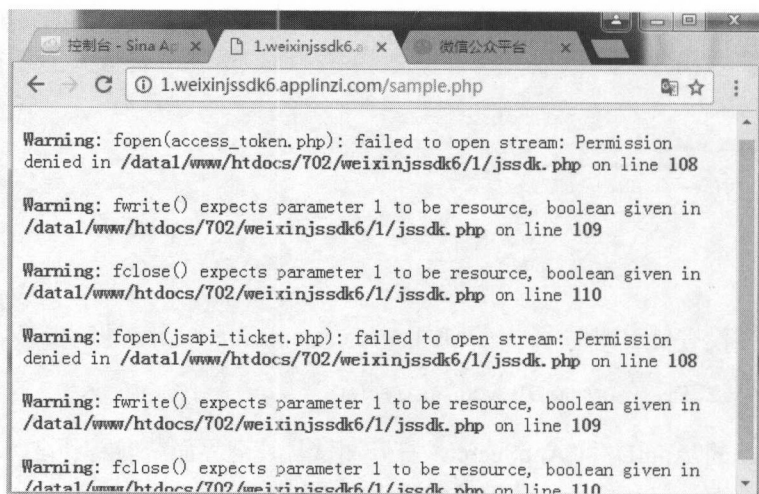


图 2.18 SAE 不支持 I/O 操作

此时，除了修改 AppID，我们对微信官方代码没有做任何修改。接下来我们就来修改代码。

2.3.4 针对 SAE 不支持读写操作的处理——Memcached

官方的 jssdk.php 中有两个函数：getJsApiTicket 和 getAccessToken。就是它们用到了 I/O 的那些函数，我们将这两个文件的内容用 Memcached 来处理。Memcached 是一个分布式内存对象缓存系统，通过在内存中缓存数据和对象来减少读取数据库的次数，从而提高动态数据库驱动网站的速度，也是键/值对的存储形式。步骤如下：

(1) 先把 getJsApiTicket 和 getAccessToken 合并为一个函数，代码如下所示。这里用 memcache_init、memcache_get、memcache_set 替换了原来的 fopen 等。

```

01 private function getJsApiTicket() {
02     $mc = memcache_init();
03     if($mc==false)
04     {
05         echo "mc init failed\n";
06     }
07     $accessToken = memcache_get($mc, "token");
08     if(empty($ACCESS_TOKEN)==true)

```



```

09      {
10
$ACC_TOKEN_URL="https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=$this->appld&secret=$this->appSecret";
11      $json_data=file_get_contents($ACC_TOKEN_URL);
12      $result=json_decode($json_data);
13      $ACC_TOKEN=$result->access_token;
14
      memcache_set($mc, "token", $ACC_TOKEN, 0, 7200);//参数：memcache 句柄、键值、
//数据、标记（默认 0）、有效期（过期会被服务器清除）
15      }
16      $mct = memcache_init();
17      if($mct==false)
18      {
19          echo "mc init failed\n";
20      }
21      $TICKET = memcache_get($mct, "ticket");
22      if(empty($TICKET)==true)
23      {
24          $url = "https://api.weixin.qq.com/cgi-bin/ticket/getticket?type=jsapi&access_token=$accessToken";
25          $res = json_decode($this->httpGet($url));
26          $ticket = $res->ticket;
27          memcache_set($mct, "ticket", $ticket, 0, 7200);//参数：memcache 句柄、键值、数
//据、标记（默认 0）、有效期（过期会被服务器清除）
28      }
29      return $TICKET;
30      }

```

(2) 因为原来的 jsapi_ticket.php 和 access_token.php 文件只是缓存数据，所以我们也将其改为 JSON 文件 access_token.json 和 jsapi_ticket.json，内容都去掉原来的第一行“<?php exit();?>”。

更改后的目录结构如图 2.19 所示。

名称	修改日期
access_token.json	2017/1/22 星期...
jsapi_ticket.json	2017/1/22 星期...
jssdk.php	2017/1/22 星期...
sample.php	2017/1/22 星期...

图 2.19 更改后的目录

(3) 代码更改完了, 我们需要在 SAE 项目中打开 Memcached, 找到自己的应用列表, 单击应用名称 (如本例 weixinjssdk6), 打开应用界面, 如图 2.20 所示。

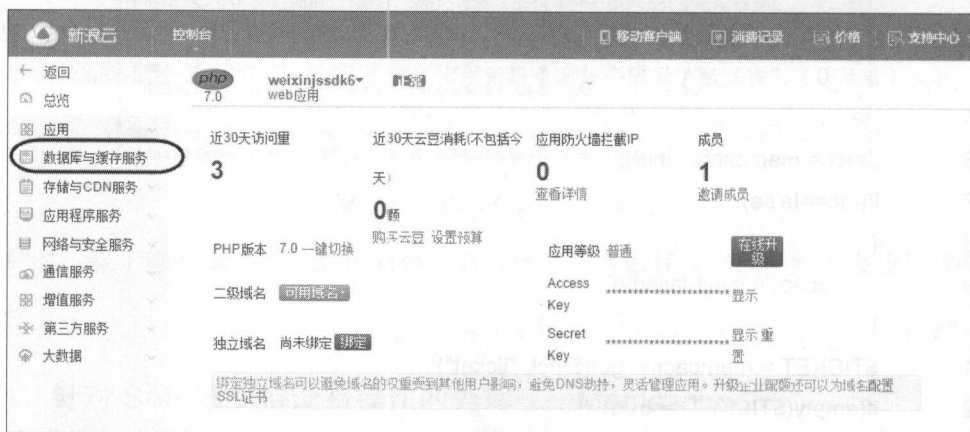


图 2.20 应用界面

(4) 单击左侧列表中的“数据库与缓存服务”|“Memcached”, 就会看到 SAE 支持的 Memcached。单击“创建 Memcached”按钮, 会弹出创建界面, 如图 2.21 所示。这里选择最小的容量就可以, 单击“确定”按钮。

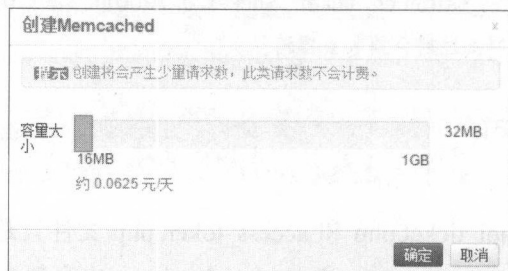


图 2.21 创建 Memcached

(5) 创建好 Memcached 后, 因为我们的文档都更新了, 所以需要再次更改远程项目, 还是使用如下代码:


```
$ git add -- all
$ git commit -m "update sample"
$ git push sae master:1
```

(6) 此时，在浏览器中输入：<http://1.weixinjssdk6.applinzi.com/sample.php>，就可以打开一个空白页面，不再有错误提示了，如图 2.22 所示。

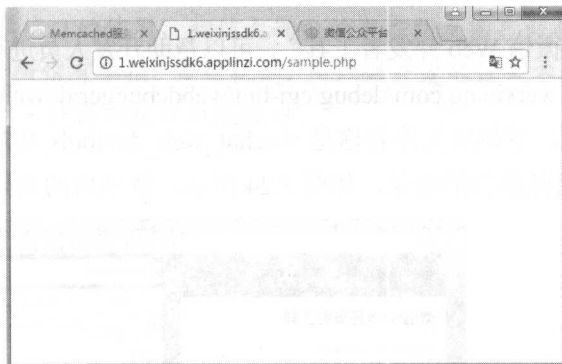


图 2.22 配置服务器成功

2.3.5 在公众号中测试效果

JSSDK 是以网页形式存在的内容，而不是微信中常看到的语音、文字或视频。既然是网页，我们就需要插入一个网址在公众号内。目前常用的插入网址的方法有 3 种：

- 在阅读原文中指定网址（任意订阅号或服务号）；
- 在自定义菜单中指定网址（必须是认证的订阅号或服务号）；
- 为图片指定网址（必须是认证的订阅号或服务号）。

既然第 1 种方法不需要认证，那本节就用它来测试。

登入公众号后台，选择“群发功能”|“新建图文消息”，这个时候主题和内容可以随意写，但是要选中“原文链接”复选框，如图 2.23 所示，输入上一小节测试好的地址。

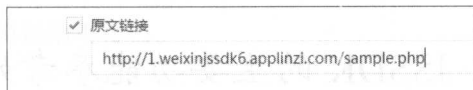


图 2.23 添加网址

在微信订阅号里找到刚才发布的那条消息，单击“阅读原文”链接会显示一个空白页

面，这样整个测试过程就完成了。

将来我们就可以在 sample.php 中增加内容、调用接口，而基本上不用修改其他文件的内容了。

2.4 JSSDK 的调试工具

目前微信提供了“微信 Web 开发者工具”，可以帮助开发人员进行开发和调试，工具下载的地址是：<https://mp.weixin.qq.com/debug/cgi-bin/webdebugger/download?from=mpwiki&os=x64>（Windows64 位）。下载后文件名称是 wechat_web_devtools_0.7.0_x64.exe。目前最新版本是 0.70，需要使用微信扫描登录，如图 2.24 所示。登录后的效果如图 2.25 所示。



图 2.24 使用微信扫描登录

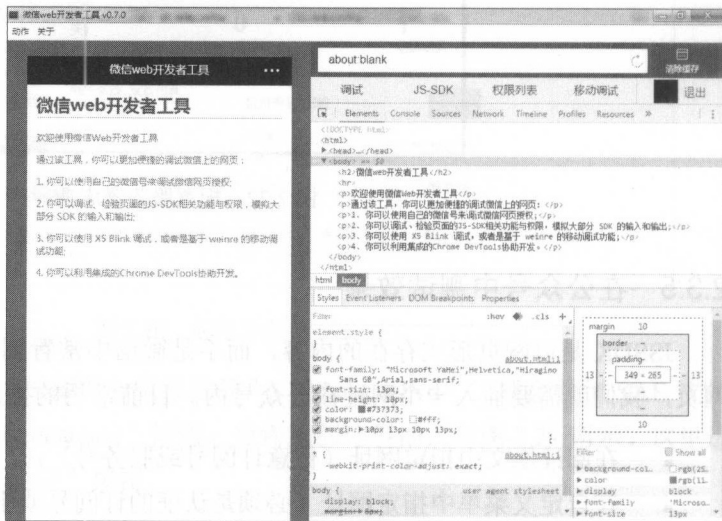


图 2.25 登录后的效果

因为要使用这个调试工具，我们还需要在后台与本人的微信号绑定，而很多人并没有公众号，所以这个工具的使用并不方便。由于网页开发人员已经习惯用 Chrome 等浏览器直接调试，因此这个工具的使用我们不再详细介绍，详细内容可参考微信的官方文档。

2.5 JSSDK 的主要功能参考列表

JSSDK 能有这么大的威力搅动 HTML 5 业界，是因为它开放了一堆 API。下面我们一起来看看。

(1) 分享接口

- 获取“分享到朋友圈”按钮点击状态及自定义分享内容接口。
- 获取“分享给朋友”按钮点击状态及自定义分享内容接口。
- 获取“分享到QQ”按钮点击状态及自定义分享内容接口。
- 获取“分享到腾讯微博”按钮点击状态及自定义分享内容接口。

小贴士：关于分享接口的作用，相信大部分读者在朋友圈已经领教过了。分享接口可以帮助商家宣传。由于微信有举报、审核机制，所以做开发的读者不要发一些敏感信息（你懂的），以免导致违规，使得此接口权限被封。

(2) 图像接口

- 拍照或从手机相册中选图接口。
- 预览图片接口。
- 上传图片接口。
- 下载图片接口。

(3) 音频接口

- 开始录音接口。
- 停止录音接口。
- 监听录音自动停止接口。
- 播放语音接口。
- 暂停播放接口。
- 停止播放接口。
- 监听语音播放完毕接口。
- 上传语音接口。
- 下载语音接口。

(4) 智能接口

- 识别音频并返回识别结果接口。

(5) 设备信息

- 获取网络状态接口。

(6) 地理位置

- 使用微信内置地图查看位置接口。
- 获取地理位置接口。

(7) 摇一摇周边

- 开启查找周边 ibeacon 设备接口。
- 关闭查找周边 ibeacon 设备接口。
- 监听周边 ibeacon 设备接口。

(8) 界面操作

- 隐藏右上角菜单接口。
- 显示右上角菜单接口。
- 关闭当前网页窗口接口。
- 批量隐藏功能按钮接口。
- 批量显示功能按钮接口。
- 隐藏所有非基础按钮接口。
- 显示所有功能按钮接口。

(9) 微信扫一扫

- 调起微信扫一扫接口。

(10) 微信小店

- 跳转微信商品页接口。

(11) 微信卡券

- 调取适用于门店的卡券列表并获取用户选择列表。
- 批量添加卡券接口。
- 查看微信卡包中的卡券接口。

(12) 微信支付

- 发起一个微信支付请求。

2.6 一些其他基础入门必知的知识点

由于 JSSDK 的运行还依赖于一些其他的语言或环境，所以在具备 JavaScript 的基本功之外，还要求会点其他知识：

- HTML/HTML5。HTML 是超文本标记语言，HTML 5 是 2014 年更新的标准，大部分浏览器都支持，尤其是在移动智能设备上的应用更为广泛。微信的 JSSDK 也是基于 HTML 5 的最新特性开发的。
- CSS/CSS 3。层叠样式表（Cascading Style Sheets）可以控制样式和布局，CSS 3 是最新的 CSS 标准。现在的 CSS 3 有很多强大的功能（丰富的选择器、径向渐变、拉伸、旋转等）。
- 基本的后台知识。由于在实际的开发项目中，会牵涉到不同的需求，JS 虽然强大，但是有时候也需要后台配合。例如，在部署 JSSDK 时，就需要一些基本的 PHP 脚本缓存令牌、临时票据。当然后台语言不一定必须是 PHP，也可以用 Node.js、Java 等。
- 一些高效开发的框架，如 jQuery.js 和 Zepto.js。很多前端开发人员已经习惯了用 jQuery 来操作 DOM，而 Zepto.js 更适合移动端的网页操作，如各种 touch 事件，所以微信在“微信网页样式库”中也默认使用了 Zepto.js。

2.7 本章小结

本章的第一个目的是让读者了解 JSSDK 开发的网页到底是否适合自己的项目。第二个目的是搭建一个 JSSDK 的开发环境，并设计好所需要的一些脚本库和文件。2.2.3 节内容比较多，从大的步骤上来说，整个搭建环境有 5 步；从小的文件上来说，各种 PHP 脚本和 JS 库文件都需要我们提前准备好。如果本章的环境没有搭建好，那么后面的公众号功能测试就不会成功，所以本章是后面所有章节的基石。读者尤其要注意的是，如果已经有自己认证的公众号，可以参考 2.2 节的搭建部署方式；如果读者只是学习，还没有自己的认证公众号，则需要参考 2.3 节使用测试号部署的方式。

第 3 章 JSSDK 网页开发的基础接口和分享接口

在学习了第 2 章之后，相信读者对 JSSDK 网页环境的搭建已经有了基本掌握，在有了基础环境之后，后面的 API 学习，就变得相对容易一些了。以后的讲解会省略配置 JSSDK 的代码及说明，直接介绍与每章主题相关的代码，如果读者对环境还不熟悉，可以翻阅第 2 章。

3.1 基础接口的作用

微信公众号的基础接口，主要是为 JSSDK 提供一些基础的检测服务。

3.1.1 任何开发都要从基础接口开始

有了基础接口，可以保证使用 JSSDK 接口的开发者更合理地依据项目需求，调整程序逻辑。例如要分享一些图片或者录制一些音频，由于您的公众号权限不够或其他原因造成无法调用 JSSDK 的某些 API，那么基础接口就有了用武之地了，因此任何开发都要从基础接口开始。

3.1.2 判断当前客户端版本是否支持指定 JS 接口

目前微信的 JSSDK 仅提供了一种基础接口，即判断当前客户端版本是否支持指定的 JS 接口。

API 名字：checkJsApi。

API 调用，示例代码如下：

```
01 wx.checkJsApi({  
02     jsApiList: [], //需要检测的 JS 接口列表，所有 JS 接口列表见附录 B  
03     success: function(res) {  
04         //成功调用后的回调结果  
05     }  
06 });
```

为了更好地在实际项目中使用，结合第 2 章的微信测试号环境，笔者修改 sample.php 文件中 wx.ready 的内容：

```
wx.ready(function () {  
    // 在这里调用 API  
    wx.checkJsApi({  
        jsApiList:['chooseImage'],  
        success: function(res) {  
            alert(JSON.stringify(res));  
        }  
    });  
});
```

这里我们测试 chooseImage 接口是否支持，然后输出反馈信息，最后在微信中测试这个网页的输出效果如图 3.1 所示。这里只测试了一个接口，如果测试多个，使用逗号间隔。反馈信息中，true 表示当前接口可用。



图 3.1 checkJsApi 测试效果

3.2 分享接口的作用

3.2.1 分享接口有什么好处

要想知道微信的 JSSDK “分享接口”有什么好处，那么首先要知道“分享”有什么好处。

“分享”就是与他人分享一些享受、使用心得等，可以加速一件事的传播，增加曝光率。最近，好多脍炙人口的网络流行语、有趣的游戏、搞笑的段子都是基于微信传播（分享）、扩散（分享）的。还有一些超级牛的商业文章，瞬间铺满全网，为公司带来大量利润，也是微信的功劳。当然，也有通过微信做公益事业的。

可见，分享的好处不言自明，因此开发者需要掌握 JSSDK 的分享接口，分享接口就是帮助开发者实现这些信息的分享。

3.2.2 获取微信的分享接口

目前，微信的 JSSDK 有 5 种分享 API，分别是：

- 分享到朋友圈。
- 分享给朋友。
- 分享到 QQ。
- 分享到腾讯微博。
- 分享到 QQ 空间

这几种 API 的调用方式类似，下面统一给出。

（1）分享到朋友圈

API 名字：onMenuShareTimeline。

API 调用，示例代码如下：

```
01 wx.onMenuShareTimeline({
02     title: "",          // 分享标题
03     link: "",           // 分享链接
04     imgUrl: "",         // 分享图标
05     success: function () {
06         // 用户确认分享后执行的回调函数
```



```

07     },
08     cancel: function () {
09         // 用户取消分享后执行的回调函数
10     }
11 });

```

(2) 分享给朋友

API 名字: `onMenuShareAppMessage`。

API 调用, 示例代码如下:

```

01 wx.onMenuShareAppMessage({
02     title: "",           // 分享标题
03     desc: "",           // 分享描述
04     link: "",           // 分享链接
05     imgUrl: "",         // 分享图标
06     type: "",           // 分享类型, music、video 或 link, 不填默认为 link
07     dataUrl: "",        // 如果 type 是 music 或 video, 则需提供数据链接, 默认为空
08     success: function () {
09         // 用户确认分享后执行的回调函数
10     },
11     cancel: function () {
12         // 用户取消分享后执行的回调函数
13     }
14 });

```

(3) 分享到 QQ

API 名字: `onMenuShareQQ`。

API 调用, 示例代码如下:

```

01 wx.onMenuShareQQ({
02     title: "",           // 分享标题
03     desc: "",           // 分享描述
04     link: "",           // 分享链接
05     imgUrl: "",         // 分享图标

```

```
06      success: function () {  
07          // 用户确认分享后执行的回调函数  
08      },  
09      cancel: function () {  
10          // 用户取消分享后执行的回调函数  
11      }  
12  });
```

(4) 分享到腾讯微博

API 名字: onMenuShareWeibo。

API 调用, 示例代码如下:

```
01  wx.onMenuShareWeibo({  
02      title: "",           // 分享标题  
03      desc: "",           // 分享描述  
04      link: "",           // 分享链接  
05      imgUrl: "",        // 分享图标  
06      success: function () {  
07          // 用户确认分享后执行的回调函数  
08      },  
09      cancel: function () {  
10          // 用户取消分享后执行的回调函数  
11      }  
12  });
```

(5) 分享到 QQ 空间

API 名字: onMenuShareQQZone。

API 调用, 示例代码如下:

```
01  wx.onMenuShareQQZone({  
02      title: "",          // 分享标题  
03      desc: "",          // 分享描述  
04      link: "",          // 分享链接  
05      imgUrl: "",        // 分享图标
```

```

06      success: function () {
07          // 用户确认分享后执行的回调函数
08      },
09      cancel: function () {
10          // 用户取消分享后执行的回调函数
11      }
12  });

```

3.3 实例：微信分享一个网页到朋友圈

(1) 首先，在 sample.php 中更改 wx.config 配置参数 jsApiList，代码如下：

```

01      jsApiList: [ // 其他代码略
02          "onMenuShareTimeline",
03          "onMenuShareAppMessage",
04          "onMenuShareQQ",
05          "onMenuShareWeibo",
06          "onMenuShareQZone"
07      ] // 必填，需要使用的 JS 接口列表，所有 JS 接口列表见附录 B

```

(2) 还是在 sample.php 文件中修改 wx.ready 内容，增加如下代码：

```

01      wx.ready(function () {
02          var title = "HTML5 外包，HTML5 外包，HTML5 是我们的生活，值得信赖的 HTML5
解决方案提供商！";
03          link = "http://www.html5waibao.com",
04          imgUrl = "http://www.html5waibao.com/images/logo_35.png",
05          desc = "html5 外包,HTML5 外包,html5 外包,html5 活,html5 手机网站",
06          success = function(){
07              alert("分享成功回调");
08          },
09          cancel = function(){
10              alert("分享失败回调");
11          };
12          wx.onMenuShareTimeline({ // 分享到朋友圈
13              title: title, // 分享标题

```

```
14         link: link,          // 分享链接
15         imgUrl: imgUrl,      // 分享图标
16         success: function () {
17             success();
18
19         },
20         cancel: function () {
21             cancel();
22
23         }
24     });
25     wx.onMenuShareAppMessage({
26         title: title,          // 分享标题
27         desc: desc,            // 分享描述
28         link: link,            // 分享链接
29         imgUrl: imgUrl,        // 分享图标
30         type: "link",          // 分享类型，music、video 或 link，不填默认为 link
31         dataUrl: "",           // 如果 type 是 music 或 video，则需提供数据链接，默认为空
32         success: function () {
33             success();
34
35         },
36         cancel: function () {
37             cancel();
38
39         }
40     });
41     wx.onMenuShareQQ({
42         title: title,          // 分享标题
43         desc: desc,            // 分享描述
44         link: link,            // 分享链接
45         imgUrl: imgUrl,        // 分享图标
46         success: function () {
47             success();
48
49         },
50         cancel: function () {
```



```

48         cancel();
49     }
50 });
51 wx.onMenuShareWeibo({
52     title: title,      // 分享标题
53     desc: desc,        // 分享描述
54     link: link,        // 分享链接
55     imgUrl: imgUrl,    // 分享图标
56     success: function () {
57         success();
58     },
59     cancel: function () {
60         cancel();
61     }
62 });
63 wx.onMenuShareQZone({
64     title: title,      // 分享标题
65     desc: desc,        // 分享描述
66     link: link,        // 分享链接
67     imgUrl: imgUrl,    // 分享图标
68     success: function () {
69         success();
70     },
71     cancel: function () {
72         cancel();
73     }
74 });
75 });

```

【代码解释】

- 因为每个 API 都会用到 title、desc 这些变量，所以第 2~11 行先定义了这些变量，其中也包括两个函数。
- 5 个接口的内容基本类似，这里全部给出。

为了测试方便，我们在 sample.php 的 body 中添加两行内容：

```
<h1 style="font-size: 40px">:)</h1>
<b style="font-size: 20px">分享接口的作用！</b>
```

代码书写完毕后，不要忘记更新到远程服务器。

打开手机看到如图 3.2 所示的 UI。点击右上角的分享到朋友圈选项，会看到如图 3.3 所示的 UI。如果分享成功会弹出对应的提示，如图 3.4 所示。

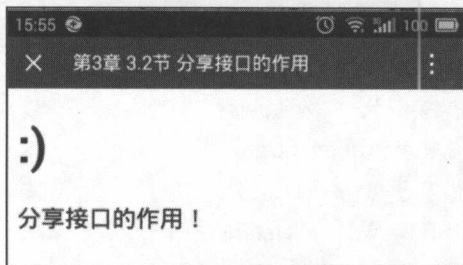


图 3.2 测试用例分享界面



图 3.3 分享到朋友圈 UI

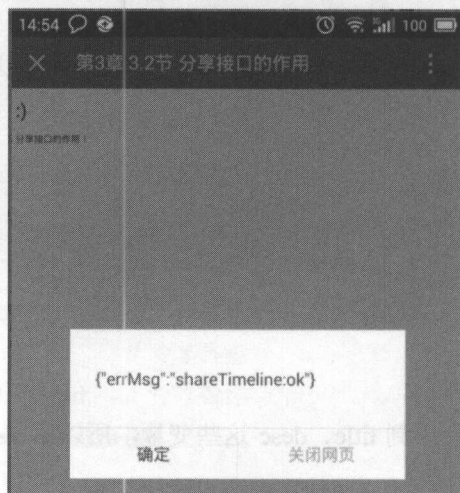


图 3.4 分享到朋友圈成功提示

3.4 常见问题

1. 无法调用分享接口

在实际的项目开发中偶尔会遇到分享接口无法调用的问题，例如无法分享到朋友圈、无法分给 QQ 好友等。

这些类似的问题一般是由以下几种情况造成的：

- 公众号本身没有权限。解决办法：向官方申请获取权限。
- JSSDK 的初始化环境失败。解决办法：检查 `access_token` 与 `jsapi_ticket` 是否可以使用、是否过期；配置 JSSDK 的参数是否正确。
- 微信 JSSDK 的 API 不稳定。解决办法：多次请求 API；寻求官方的技术支持，解决 API 不稳定的 Bug。
- JSSDK 没有初始化完毕，就执行分享操作。
- 初始化 JSSDK 的时候，没有在“`jsApiList`”中配置 API 参数。

第 4 章 JSSDK 的图像处理

相信本书的读者都使用过微信，微信不仅仅是一堆文字的相互传播，和图形相关的需求也不少，例如：

- 在个人设置选项中，需要自定义头像。
- 在发型适配的项目中，需要用户的发型图片。
- 现在比较流行的人脸识别。
- 二维码的处理。

以上这些形形色色的项目，本质上都需要进行图像处理，因此 JSSDK 中的图像 API 就变得比较重要了，本章就来学习图像 API。

4.1 认识图像接口

JSSDK 的图像 API 目前总共有 4 种，基本覆盖了图像的全部处理业务：

- 拍照或从手机相册中选图接口；
- 预览图片接口；
- 上传图片接口；
- 下载图片接口。

4.1.1 从手机相册中选图

第 1 个 API 的名称是“chooseImage”，主要用于获取图片。拍照或从相册中选图，都交给这个 API 完成，调用方式示例如下：


```

01 wx.chooseImage({
02     success: function (res) {
03         var localIds = res.localIds; // 会返回选中照片的本地 ID 列表，数组形式，localId 可以
04         作为 img 标签的 src 属性显示图片
05     }
06 });

```

我们来演示一下具体的应用。在 sample.php 中，添加该接口和调用该接口，代码如下：

```

01 <style>           <!--按钮样式设计-->
02     input{
03         width: 100%;
04         padding: 0.2em;
05         background-color: #5eb95e;
06         font-size: 1.4em;
07         background-image: linear-gradient(to bottom, #62c462, #57a957);
08         background-repeat: repeat-x;
09         color: #ffffff;
10         text-align: center;
11         text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
12         border-radius: 0.3em;
13     }
14 </style>
15
16 <body>           <!--页面主体设计-->
17     <h1 style="font-size: 8em">:</h1>
18     <b style="font-size: 3em">认识图像接口！</b><br /><br />
19     <input type="button" value="拍照或从手机相册中选图接口" id="chooseImage" />
20     <br/><br />
21 </body>
22 <script src="http://res.wx.qq.com/open/js/jweixin-1.0.0.js"></script>
23 <script src="jquery-1.11.2.min.js"></script>
24 <script>
25     wx.config({
26         debug: true,

```

```

27     appld: '<?php echo $signPackage["appld"];?>',
28     timestamp: '<?php echo $signPackage["timestamp"];?>',
29     nonceStr: '<?php echo $signPackage["nonceStr"];?>',
30     signature: '<?php echo $signPackage["signature"];?>',
31     jsApiList: [
32         "chooseImage",
33     ]
34 });
35 wx.ready(function () {
36     // 在这里调用 API
37     var chooseImageID;//拍照或从手机相册中选图接口
38     $('#chooseImage').click(function(){
39         wx.chooseImage({
40             count: 1, // 默认 9
41             sizeType: ['original', 'compressed'], //指定是原图还是压缩图
42             sourceType: ['album', 'camera'], //指定来源是相册还是相机
43             success: function (res) {
44                 chooseImageID = res.localIds[0]; // 返回选定照片的本地 ID
45                 alert('选择的图片 ID'+chooseImageID);
46             }
47         });
48     });
49 });
50 </script>

```

【代码解释】

- 第 19 行创建一个“id="chooseImage"”的“input”元素，用来做选择照片的按钮。
- 第 38~48 行为“id="chooseImage"”的元素绑定一个事件，响应选择图片 API，这里使用了 jQuery 来绑定事件，所以要在第 23 行引入 jquery-1.11.2.min.js 文件。
- 不要忘记在第 32 行添加 chooseImage 到配置表中。

单击主页中的按钮就会调用图像，如图 4.1 所示。

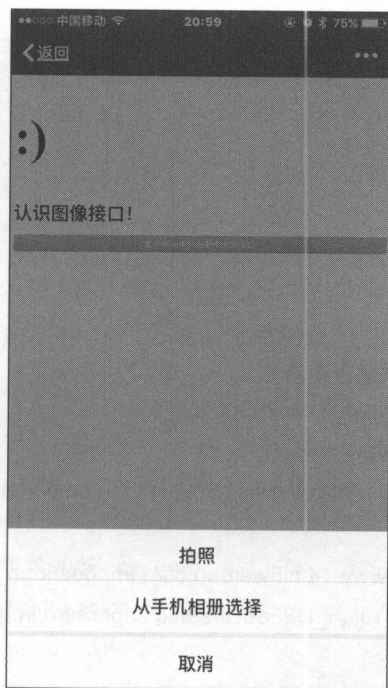


图 4.1 微信 JSSDK 选择图片 UI

4.1.2 预览图片

第 2 个 API 的名称是“previewImage”，主要用于预览图片，调用方式示例如下：

```
01 wx.previewImage({
02     current: "", // 指定当前要显示的图片链接
03     urls: []    // 数组形式，需要预览的图片链接列表
04 });
```

继续来演示，在 sample.php 中添加 previewImage 的配置、一个预览按钮、为按钮绑定事件：

```
01 <input type="button" value="预览图片接口" id="previewImage" />
02 wx.config({
03     debug: true,
04     appId: '<?php echo $signPackage["appId"];?>',
05     timestamp: <?php echo $signPackage["timestamp"];?>,
```

```

06     nonceStr: '<?php echo $signPackage["nonceStr"];?>',
07     signature: '<?php echo $signPackage["signature"];?>',
08     jsApiList: [
09         "chooseImage",
10         "previewImage"
11     ]
12 });
13 wx.ready(function () {
14     //...省略前面介绍过的选择图像接口
15     $('#previewImage').click(function(){
16         wx.previewImage({
17             current:"http://www.html5waibao.com/images/logo_35.png",
18             urls:[
19                 "http://www.html5waibao.com/images/logo_35.png",
20                 "https://ss0.bdstatic.com/5aV1bjqh_Q23odCf/static/superman/img/logo/bd_logo1_31bdc765.png"
21             ]
22         });
23     });
24 });

```

【代码解释】

- 第 1 行创建一个“id="previewImage"”的“input”元素。
- 第 15~20 行为“id="previewImage"”的元素绑定一个事件，响应预览图片 API，要注意这里绑定的是两个数值，并不像其他 API 一样有 success 方法。
- 第 10 行增加该接口的配置。

预览图片的 UI 效果如图 4.2 所示。

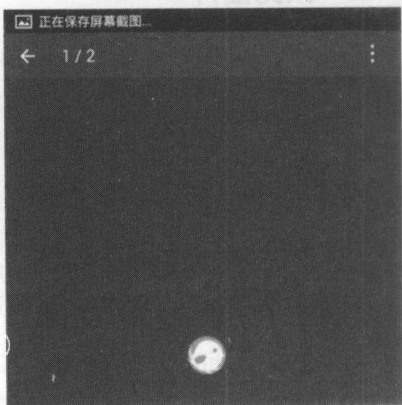


图 4.2 预览图片 UI

4.1.3 上传图片

第 3 个 API 的名称是“uploadImage”，主要用于上传图片，调用方式示例如下：


```

01  wx.uploadImage({
02      localId: "",           // 等待上传图片的本地 ID，由 chooseImage 的 API 获得
03      isShowProgressTips: 1, // 默认为 1，显示进度提示
04      success: function (res) {
05          var serverId = res.serverId; // 返回图片的服务器端 ID
06      }
07  });

```

还是在 sample.php 中添加配置、上传图片按钮、绑定事件，代码如下：

```

01  <input type="button" value="上传图片接口" id="uploadImage" />
02  wx.config({
03      debug: true,
04      appId: '<?php echo $signPackage["appId"];?>',
05      timestamp: <?php echo $signPackage["timestamp"];?>,
06      nonceStr: '<?php echo $signPackage["nonceStr"];?>',
07      signature: '<?php echo $signPackage["signature"];?>',
08      jsApiList: [
09          "chooseImage",
10          "previewImage",
11          "uploadImage"
12      ]
13  });
14  wx.ready(function () {
15      var chooseImageId; // 拍照或从手机相册中选图接口
16      // 省略前两个 API 的代码
17      $('#uploadImage').click(function () {
18          if (!chooseImageId) {
19              alert('请先选择图片，再上传！');
20              return;
21          }
22          wx.uploadImage({
23              localId: chooseImageId.toString(), // 需要上传的图片的 ID
24              isShowProgressTips: 1,           // 默认为 1，显示进度提示

```

```

25          success: function (res) {
26              uploadImageID = res.serverId;    // 返回图片的服务器端 ID
27              alert("上传的图片 ID"+uploadImageID); // 返回图片的服务器端 ID
28          }
29      });
30  });
31  });

```

【代码解释】

- 第 1 行创建一个“id="uploadImage"”的“input”元素。
- 第 17~30 行为“id="uploadImage"”的元素绑定一个事件，响应上传图片 API。另外，有时候 JSSDK 上传图片会无响应，记得要在配置“localId”的时候，用“toString()”函数转换一下数据。

本例在 JSSDK 调试模式下，会弹出如图 4.3 所示的 UI。

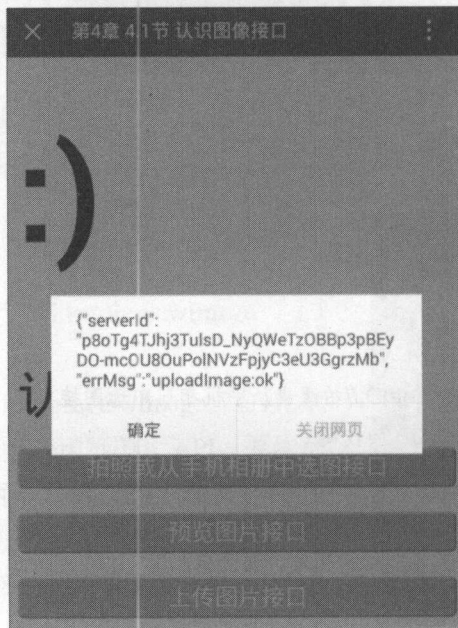


图 4.3 手机微信的 JSSDK，上传图片成功提示

提醒：微信上传图片有效期是 3 天，由于 JSSDK 的下载图片 API 不稳定，有时候不能使用，可以使用下载多媒体文件 API 下载图片，以解决此问题。

下载多媒体的 API 调用示例如下：

```

01 //格式说明：http 请求方式：GET
02 http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=ACCESS_TOKEN&media_id=
03 MEDIA_ID
04
05 //示例代码（为通过 curl 命令获取多媒体文件）
06 http://file.api.weixin.qq.com/cgi-bin/media/get?
07 access_token=eQv3HPwEFxsw8cyh5O7DjaNOoGd4d
08 -jYtG_ZtmX8ZW8np0Q5CqAox7lghNkNuiNHU8M6c9uW-YbwUYxkMywh
09 _O3LCC18jbRvcaLjQuHq8&media_id="+res.serverId; // 返回图片的服务器端 ID

```

更多信息请参考官方文档：<http://mp.weixin.qq.com/wiki/12/58bfcfabbd501c7cd77c19bd9cfa8354.html>。

4.1.4 下载图片

图像处理的第 4 个 API 就是“downloadImage”，调用方式如下：

```

01 wx.downloadImage({
02     serverId: "", // 由 uploadImage 接口获得，下载图片的服务器端 ID
03     isShowProgressTips: 1, // 显示进度提示，默认为 1
04     success: function (res) {
05         var localId = res.localId; // 图片下载后的本地 ID 资源
06     }
07 });

```

还是在 sample.php 中，添加配置、下载图片按钮、绑定事件，代码如下：

```

01 <input type="button" value="下载图片接口" id="downloadImage" />
02 wx.config({
03     debug: true,
04     appId: '<?php echo $signPackage["appId"];?>',
05     timestamp: '<?php echo $signPackage["timestamp"];?>',
06     nonceStr: '<?php echo $signPackage["nonceStr"];?>',
07     signature: '<?php echo $signPackage["signature"];?>',

```



```
08     jsApiList: [  
09         "chooseImage",  
10         "previewImage",  
11         "uploadImage",  
12         "downloadImage"  
13     ]  
14 };  
15 wx.ready(function () {  
16     //省略部分代码  
17     $('#downloadImage').click(function(){  
18         if(!uploadImageID){  
19             alert('请先上传图片，再下载！');  
20             return;  
21         }  
22  
23         wx.downloadImage({  
24             serverId:uploadImageID, // 默认为 1，显示进度提示  
25             success: function (res) {  
26                 alert("下载的图片 ID"+res.serverId);  
27             },  
28             fail:function(res){  
29                 alert(JSON.stringify(res));  
30             }  
31         });  
32     });  
33 });
```

【代码解释】

其他代码都跟前面的 API 类似，不再解释。这里要注意的是，只有选择图片、上传图片之后才能使用下载图片的 API。

4.2 实例：从手机相册中选择照片然后分享

到目前为止，已经学了3种JSSDK的API，可以做一些简单的需求了，本节讲解一个让用户在手机中选择照片然后分享的项目。

4.2.1 项目需求

需求说明：实现微信端的手机用户点击按钮选取1张图片，分享到朋友圈。

4.2.2 需求分解

通过对需求的了解，可以将其分解为：

(1) 微信端手机用户，可以使用微信的JSSDK。

(2) 选取图片，使用JSSDK的“chooseImage”，由于分享图片时本地地址无法分享，因此还需要JSSDK的“uploadImage”。

(3) 分享到朋友圈，需要JSSDK的“onMenuShareTimeline”。

综合起来，业务逻辑如图4.4所示。

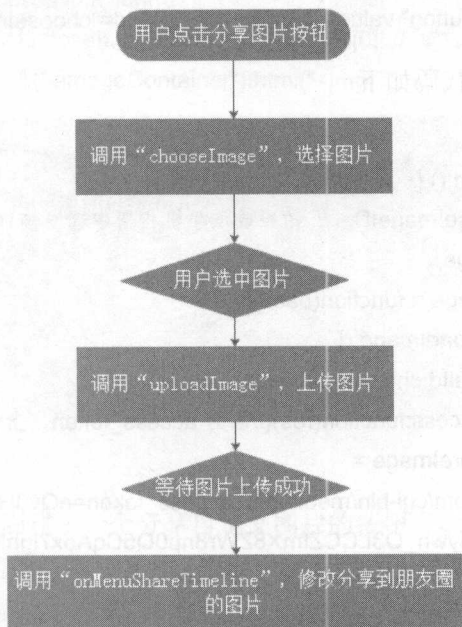


图 4.4 业务逻辑结构图

4.2.3 代码实现

首先修改 JSSDK 环境的配置文件，代码如下：

```
01  jsApiList: [ // 必填，需要使用的 JS 接口列表，所有 JS 接口列表见附录 B
02      "chooseImage",
03      "previewImage",
04      "uploadImage",
05      "onMenuShareTimeline"
06  ]
07  //其他代码略
```

依据需求，构建一个“id”等于“chooseImage”的选图按钮，以及点击该按钮后，用来显示选中图片的容器，在 sample.php 文件的 body 中增加如下代码：

```
01  <h1 style="font-size: 8em">.</h1>
02  <b style="font-size: 2em">实例：从手机相册中选照片然后分享，支持选择 1 张图片！
03  </b><br /><br />
04  <p id="imageContainer" style="text-align: center; width: 100%"></p>
05  <p id="selectImg" style="color: #5eb95e; text-align: center">没有自定义分享图片</p>
06  <input type="button" value="请选择分享图片" id="chooseImage" /><br /><br />
```

然后增加处理业务的代码如下：

```
01  wx.ready(function () {
02      var chooseImageID, // 拍照或从手机相册中选图接口
03      shareImage,
04      uploadImage = function (back) {
05          wx.uploadImage ({
06              localId: chooseImageID.toString(),
07              success: function (res) { // 临时 access_token，上传图片成功后，执行分享操作
08                  shareImage =
09                      "http://file.api.weixin.qq.com/cgi-bin/media/get?access_token=eQv3HPwEFxwsw8cyh5O7DjaNOoGd4d-jYtG_c9uW-YbwUYxkMywh_O3LCCZtmX8ZW8np0Q5CqAox7lghNkNuiNHU8M618jbRvcaLjQuHq8&media_id="+res.serverId; // 返回图片的服务器端 ID
10                  back && back();
11              }
12          });
13      }
14  });
```

```

11         });
12     },
13     shareTimeline = function(){
14         uploadImage(function(){
15             wx.onMenuShareTimeline ( { // 分享到朋友圈
16                 title: "实例：从手机相册中选照片然后分享！", // 分享标题
17                 link: 'http://weibo.com/xixinliang', // 分享链接
18                 imgUrl: shareImage, // 分享图标
19                 success: function () {
20                 },
21                 cancel: function () {
22                 }
23             }
24         });
25     });
26 };
27 $("#chooseImage").click(function(){
28     wx.chooseImage({
29         success:function(res){
30             chooseImageID = res.localIds[0]; // 返回选定照片的本地 ID 列表
31             $("#imageContainer").html("<img style='width: 30%'
src='"+chooseImageID+"'>");
32             $("#selectImg").html("已选择图片，请点击右上角分享到朋友圈按钮");
33             shareTimeline();
34         }
35     });
36 });
37
38 });

```

【代码解释】

- 在 sample.php 中，创建了一个分享自定义图片的按钮，如图 4.5 所示。点击分享按钮，调用 JSSDK 的选择图片 API，让用户选中图片，如图 4.6 所示。
- 之后，调用“uploadImage”上传图片。

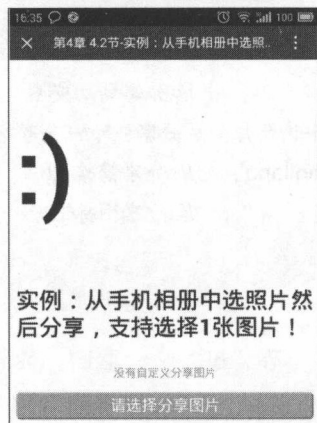


图 4.5 自定义分享图片 UI

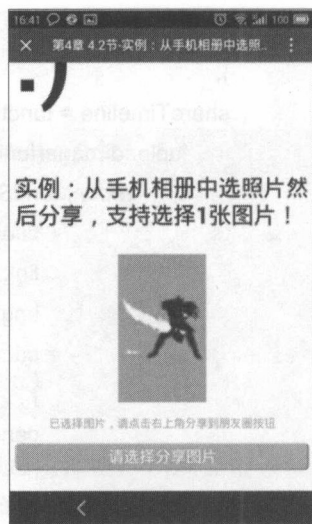


图 4.6 选择图片之后的 UI

- 上传成功之后，返回上传之后的服务端“serverId”，再调用下载多媒体的 API，将图片赋值给 JSSDK 的分享 API “onMenuShareTimeline”。

用户可以查看分享效果，如图 4.7 和图 4.8 所示。

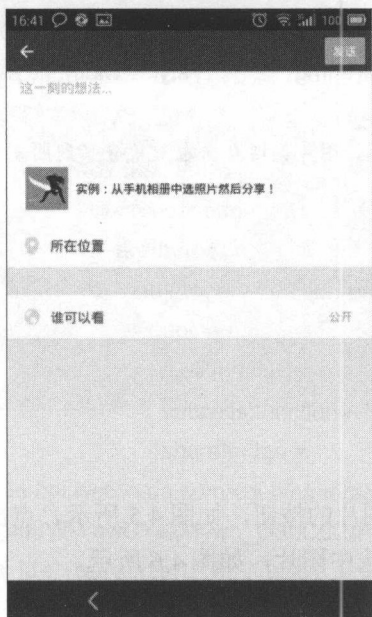


图 4.7 分享到朋友圈编辑的 UI



图 4.8 成功分享自定义图片到朋友圈

4.3 常见问题

图像接口的主要问题，除了配置环境与接口权限问题之外，主要集中在上传图片与下载图片，前面讲过，再简略回顾一下。

1. 上传图片的主要问题是无响应。

解决方法：只要记得在配置“localId”的时候，用“toString()”转一下类型即可。

2. 下载图片的主要问题是官方的 API 不稳定。

解决方法：等官方解决 API 问题；用下载多媒体文件的 API 替代。

3. require subscribe 错误。

使用微信测试号的读者如果使用多个手机测试可能会出现此问题，这主要是因为其他手机没有订阅该测试号，该错误仅测试号会出现。

第 5 章 JSSDK 的音频处理

信息的产生、传递始终是每个时代的主旋律。当今社会，移动互联网信息传播的主要表现形式是富媒体，而其中音频占的比重非常大。微信的核心功能就是 IM 音频通信，因此学习 JSSDK 的音频处理也是不可缺少的环节。

5.1 音频接口

JSSDK 的音频 API 覆盖面比较全面，包括开始录音、停止录音、监听录音自动停止、播放录音、暂停录音、停止播放、监听语音播放完毕、上传语音、下载语音。

5.1.1 开始录音

先来介绍一下第 1 个音频 API “startRecord”，主要用于“开始录音”功能。调用方式如下：

```
01 wx.startRecord();
```

在 sample.php 文件中，增加开始录音按钮的代码结构如下：

```
01 <body>
02   <h1 style="font-size: 8em">.</h1>
03   <b style="font-size: 3em">音频接口！</b><br /><br />
04   <input type="button" value="开始录音" id="startRecord" />
05 </body>
```

然后在 wx.ready 中增加响应事件，代码如下：

```
01 wx.ready(function () {
```

```

02      $("#startRecord").click(function(){ //开始录音
03          wx.startRecord();
04      });
05  });

```

最后记得在 JSSDK 的配置环境中加入 “startRecord” API 的权限。

本节中的代码逻辑比较简单易懂，只是一个开始录音按钮，只要按照格式调用即可，要想检验录音的效果，还要与后续章节结合，那么接着再一起学习停止录音接口。

5.1.2 停止录音

在真实的录音环境中，用户可能会停止或中断录音。JSSDK 为开发者也开放了一个这样的接口，叫 “stopRecord”，代码示例如下：

```

01  wx.stopRecord({
02      success: function (res) {
03          var localId = res.localId;
04      }
05  });

```

在 sample.php 文件中，增加停止录音按钮的代码结构如下：

```
<input type="button" value="停止录音" id="stopRecord" />
```

然后在 wx.ready 中绑定停止录音的事件，代码如下：

```

01  var localId,    //本地音频 ID
02  serverId;       //服务端 ID
03  //其他代码略
04      $("#stopRecord").click(function(){ //停止录音
05          wx.stopRecord({
06              success:function(res){ //停止录音成功回调
07                  localId = res.localId;
08              }
09          });
10      });

```

最后记得在 JSSDK 的配置环境中加入 “stopRecord” API 的权限。

要想听到录音结果，还要接着学习几个接口。

5.1.3 监听录音自动停止

有时候用户在录音的时候，有一些操作会不符合程序逻辑，例如录音超时，那么 JSSDK 也提供了一个“监听录音自动停止”的接口，名为 “onVoiceRecordEnd”，示例代码如下：

```
01  wx.onVoiceRecordEnd({  
02      // 录音时间超过一分钟没有停止的时候会执行 complete 回调  
03      complete: function (res) {  
04          var localId = res.localId;  
05      }  
06  });
```

增加超时调用的监听代码如下：

```
01  wx.onVoiceRecordEnd({  
02      complete: function (res) {  
03          localId = res.localId;  
04          alert("监听录音已自动停止");  
05      }  
06  });
```

最后记得在 JSSDK 的配置环境中加入 “onVoiceRecordEnd” API 的权限。

激动人心的“播放音频 API”要来了，想不想听一下自己录的音频？如果想，那就继续下一节吧。

5.1.4 播放语音

当录音完毕之后，肯定是要听听录音效果的，如果不听，这个 JSSDK 肯定没意义，播放音频的 API 是 “playVoice”，示例代码如下：

```
01  wx.playVoice({  
02      localId: " // 需要播放的音频的本地 ID，由 stopRecord 接口获得  
03  });
```

这个接口也相对简单，只要配置录音的本地 ID 就可以了。

在 sample.php 文件中，增加播放录过声音的按钮，代码结构如下：

```
<input type="button" value="播放录音" id="playVoice" />
```

然后增加播放录音的事件，代码如下：

```
01      $("#playVoice").click(function(){           //播放录音
02          wx.playVoice({
03              localId:localId
04          });
05      });
```

最后记得在 JSSDK 的配置环境中加入 “playVoice” API 的权限。

学到这，你是不是可以体验一把自己的成果，点击按钮听到自己的录音了？如果不可以请检查以下问题：

- 对应接口的配置是否填写正确。
- 是否是在移动端微信中的浏览器运行的实验。
- JSSDK 初始化是否完成。

运行 UI 如图 5.1 所示。

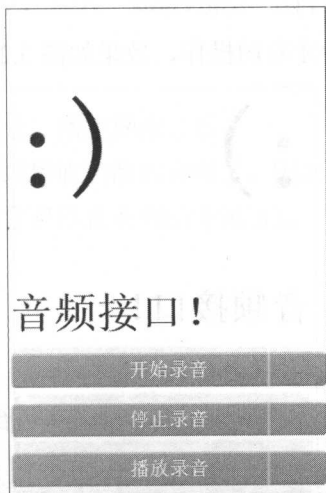


图 5.1 录音并播放的 UI

5.1.5 暂停播放

在音频播放的过程中，有时候需要暂停播放的进程，那么 JSSDK 的“pauseVoice”就有用武之地了，示例代码如下：

```
01 wx.pauseVoice({  
02     localId: "// 需要暂停的音频的本地 ID，由 pauseVoice 接口获得  
03 });
```

与播放音频类似，这个接口也比较简单，只要配置录音的本地 ID 就可以了。

sample.php 文件中，增加暂停播放的按钮，代码结构如下：

```
<input type="button" value="暂停播放" id="pauseVoice" />
```

然后在增加暂停播放的事件，代码如下：

```
01     $("#pauseVoice").click(function(){           //暂停播放  
02         wx.pauseVoice({  
03             localId:localId  
04         });  
05     });
```

最后在 JSSDK 的配置环境中，加入“pauseVoice”API 的权限。

【代码解释】

暂停录音是在播放录音之后才有的操作，效果如图 5.2 所示。

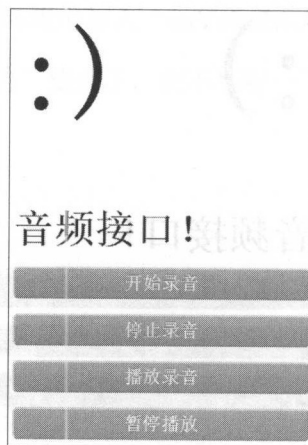


图 5.2 暂停播放按钮

5.1.6 停止播放

与暂停不同的是，停止是终止播放，不能继续的意思，JSSDK 提供了一个接口叫“stopVoice”，示例代码如下：

```
01 wx.stopVoice({
02     localId: " // 需要停止的音频的本地 ID，由 stopRecord 接口获得
03 });
```

读者现在对音频接口很熟悉了吧，这个也是配置录音的本地 ID 就可以了。

在 sample.php 文件中，增加停止播放的按钮，代码结构如下：

```
<input type="button" value="停止播放" id="stopVoice" />
```

然后增加停止播放的事件，代码如下：

```
01 //其他代码略
02 $("#stopVoice").click(function(){ //停止播放
03     wx.stopVoice({
04         localId:localId
05     });
06 });
```

最后在 JSSDK 的配置环境中，加入“stopVoice”API 的权限。

【代码解释】

- 停止播放的操作放在录音、播放操作之后。
- 停止代表终结，不能继续播放当前的音频了，因此有一个成功停止音频之后的回调函数，返回的资源列表含有停止音频的本地 ID。

5.1.7 监听语音播放完毕

除了人为地停止音频之外，音频也是有时长的，当一段音频播放完毕之后，就需要 JSSDK 的另一个 API 了，监听语音播放完毕的 API 叫“onVoicePlayEnd”，示例代码如下：

```
01 wx.onVoicePlayEnd({
02     success: function (res) {
03         var localId = res.localId; // 返回音频的本地 ID
```



```
04    }
05  });
```

这个接口需要配置音频播放完毕之后的回调。

增加监听语音播放完毕的处理回调函数，代码如下：

```
01  //其他代码略
02    wx.onVoicePlayEnd({
03      success: function (res) {
04        console.log("语音播放完毕，localId 是: "+res.localId);
05      }
06    });
```

最后在 JSSDK 的配置环境中，加入“onVoicePlayEnd”API 的权限。

5.1.8 上传语音

估计很多读者都知道现在大数据非常火，大数据的一个核心就是要有数据源，才能进行分析。另外，在很多项目需求中，都需要用户语音上传功能，例如变声应用、语音 IM 工具等。

在微信的 JSSDK 中，上传语音也是接口之一，名字叫“uploadVoice”，示例代码如下：

```
01  wx.uploadVoice({
02    localId: "", // 需要上传的音频的本地 ID，由 uploadVoice 接口获得
03    isShowProgressTips: 1, // 默认为 1，显示进度提示
04    success: function (res) {
05      var serverId = res.serverId; // 返回音频的服务器端 ID
06    }
07  });
```

这个接口需要配置本地的音频 ID，也需要一个回调函数，用来处理上传成功后的业务逻辑。

在 sample.php 文件中，增加上传音频的按钮，代码结构如下：

```
<input type="button" value="上传语音" id="uploadVoice" />
```

然后在增加上传音频的事件，代码如下：

```

01      //其他代码略
02      $("#uploadVoice").click(function(){      // 上传语音
03          wx.uploadVoice({
04              localId:localId,
05              success:function(res){
06                  serverId = res.serverId;
07                  console.log("上传语音成功，serverId: "+res.serverId);
08              }
09          });
10      });

```

最后在 JSSDK 的配置环境中，加入“uploadVoice”API 的权限。

【代码解释】

- 语音上传的主要配置参数是本地音频 ID，音频 ID 是录音生成的，音频上传成功之后，在 JSSDK 调试模式下，会有弹框提醒，如图 5.3 所示。
- 微信上传的音频，服务器会保持 3 天，并且可以用多媒体的 API 音频，和上一章的图片 API 类似，参考地址为：<http://mp.weixin.qq.com/wiki/12/58bfcfabbd501c7cd77c19bd9cfa8354.html>。

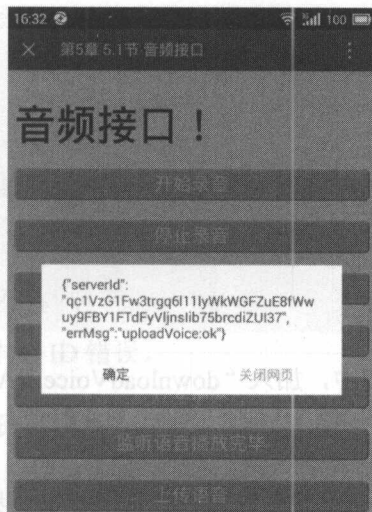


图 5.3 JSSDK 上传语音成功提示

5.1.9 下载语音

那么，我们如何将成功上传到服务器的音频下载下来呢？除了多媒体链接下载，还有其他方式吗？答案是肯定的，JSSDK 的音频下载 API “downloadVoice” 就是这样的存在，官网示例代码如下：

```
01 wx.downloadVoice({
02     serverId: "",           // 需要下载的音频的服务器端 ID，由 downloadVoice 接口获得
03     isShowProgressTips: 1,  // 默认为 1，显示进度提示
04     success: function (res) {
05         var localId = res.localId; // 返回音频的本地 ID
06     }
07 });
```

这个接口需要配置的不是本地的音频 ID，而是服务器端的 ID，当然也需要一个回调函数，用来处理下载成功后的业务逻辑。

在 sample.php 文件中，增加下载语音的按钮，代码结构如下：

```
<input type="button" value="下载语音" id="downloadVoice" /><br /><br />
```

然后增加下载音频的事件，代码如下：

```
01 //其他代码略
02 $("#downloadVoice").click(function(){ //下载语音
03     wx.downloadVoice({
04         serverId:serverId,
05         success:function(res){
06             console.log("下载语音成功，localId: "+res.localId);
07         }
08     });
09 });
```

最后在 JSSDK 的配置环境中，加入 “downloadVoice” API 的权限。

【代码解释】

语音下载的主要配置参数是上传 API 返回的服务端的音频 ID，与上传音频类似，音频下载成功之后，在 JSSDK 调试模式下，会有弹框提醒，如图 5.4 所示。

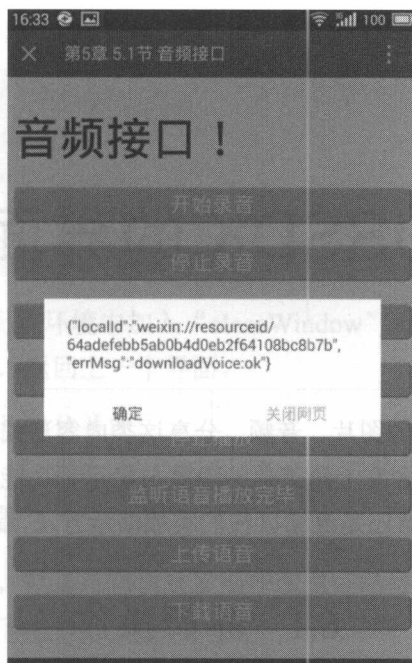


图 5.4 JSSDK 下载音频成功提示

5.2 常见问题

在实际开发中，会遇到各种各样的项目问题或疑问，在此列举一二，并给出一些解决方案。

1. 录制的 API 没有声音，怎么办？

解决方案：首先检查有没有开发权限，然后检查设备或者微信版本，最后查看 JS 代码是否有写错的地方。

2. 播放的声音不对，怎么办？

解决方案：这个一般是本地 ID 错误。

3. 下载不了音频，怎么回事？

解决方案：服务 ID 错误、权限问题、设备问题、JS 代码 Bug、JSSDK 不稳定。

第 6 章 JSSDK 的界面操作

JSSDK 除了能控制微信的图片、音频、分享这类内容形式的 API 之外，还可以控制微信的界面。这些界面操作包含右上角菜单的隐藏与显示、功能按钮的批量显示与隐藏，还有关闭当前网页窗口等。本章还是继续以往的实战演示方法，逐步介绍这些界面操作 API。

6.1 界面操作

在最新版 JSSDK 中，微信保留了 5 种界面的自定义操作，包括关闭当前网页、批量隐藏功能按钮、批量显示功能按钮、隐藏所有非基础按钮和显示所有功能按钮，主要是微信内右上角的菜单操作，以便开发者在实际项目中使用。最早版本中有“隐藏/显示”右上角菜单，但是在最新版本中去掉了这个功能，也就是说，右上角菜单无法再隐藏。

6.1.1 关闭当前网页窗口

有的时候需要终结用户的操作，重新打开网页或做其他操作，JSSDK 提供了一个实用的 API，叫“closeWindow”，官方示例代码如下：

```
wx.closeWindow();
```

在 sample.php 文件中，增加“关闭当前窗口”按钮的代码结构如下：

```
01 <body>
02     <h1 style="font-size: 8em">.</h1>
03     <b style="font-size: 3em">界面操作接口！</b><br /><br />
04     <input type="button" value="关闭当前网页窗口" id="closeWindow" />
05 </body>
```


然后在 wx.ready 中，增加“关闭当前网页窗口”的响应事件，代码如下：

```
01      $("#closeWindow").click(function(){           //关闭当前网页窗口接口
02          wx.closeWindow ({
03              closeWindow:true
04          });
05      });
```

最后记得在 JSSDK 的配置环境中加入“closeWindow”API 的权限。当用户单击该按钮时，一般是关闭当前窗口，返回上一个界面。

注意：关闭窗口并不是把微信关闭。

6.1.2 批量隐藏功能按钮

所谓批量操作功能按钮，就是可以自定义操作配置好的按钮。第 1 个是“批量隐藏功能按钮”，官方示例代码如下：

```
01  wx.hideMenuItems({
02      menuList: []
03  });
```

在 sample.php 文件中，增加“批量隐藏功能按钮”的代码结构如下：

```
01  <body>
02      <h1 style="font-size: 8em">:</h1>
03      <b style="font-size: 3em">界面操作接口！</b><br /><br />
04      <input type="button" value="批量隐藏功能按钮" id="hideMenuItems" />
05  </body>
```

然后增加“批量隐藏功能按钮”的响应事件，代码如下：

```
01      $("#hideMenuItems").click(function(){ //批量隐藏
02          wx.hideMenuItems({
03              menuList:[
04                  "menuItem:share:timeline", //隐藏分享到朋友圈
05                  "menuItem:share:qq"       //隐藏分享到 QQ
06              ]
07          });
```


08

});

最后记得在 JSSDK 的配置环境中加入“hideMenuItems”API 的权限。

【代码解释】

- “hideMenuItems”只能隐藏“传播类”和“保护类”按钮，所有 menu 项请参考附录 C。
- 本节的测试例子，将“分享到朋友圈”与“分享到 QQ”进行了隐藏，结果如图 6.1 所示。本例中读者一定要单击按钮后才会执行“隐藏”操作，如果要默认一打开界面就隐藏，可将上述操作放到 window.onload 事件中。

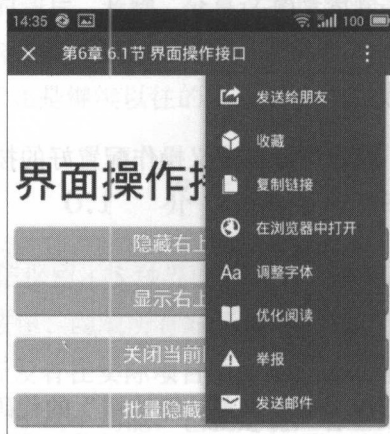


图 6.1 隐藏“分享到朋友圈”与“分享到 QQ”菜单

6.1.3 批量显示功能按钮

在 JSSDK 中，有批量隐藏的 API，当然就有批量显示的 API，这样才符合事物的辩证原理，符合市场的规则，其名字叫“showMenuItems”，官方示例代码如下：

```
01 wx.showMenuItems({
02     menuList: [] // 要显示的菜单项，所有 menu 项见附录 C
03 });
```

在 sample.php 文件中，增加“批量显示功能按钮”的代码结构如下：

```
01 <body>
02     <h1 style="font-size: 8em">.</h1>
```

```

03      <b style="font-size: 3em">界面操作接口! </b><br /><br />
04      <input type="button" value="批量显示功能按钮" id="showMenuItems" />
05  </body>

```

然后增加“批量显示功能按钮”的响应事件，代码如下：

```

01      $("#showMenuItems").click(function(){ //批量显示
02          wx.showMenuItems({
03              menuList:[
04                  "menuItem:share:timeline", //显示分享到朋友圈
05                  "menuItem:share:qq"      //显示分享到 QQ
06              ]
07          });
08      });

```

最后记得在 JSSDK 的配置环境中加入“showMenuItems”API 的权限。

【代码解释】

“批量显示功能”的 API 主要是和“批量隐藏”的 API 组合使用，当您测试的时候，需要和上一节学的内容放在一起测试，本节中将上一节中隐藏的“分享到朋友圈”与“分享到 QQ”菜单进行显示，效果如图 6.2 所示。

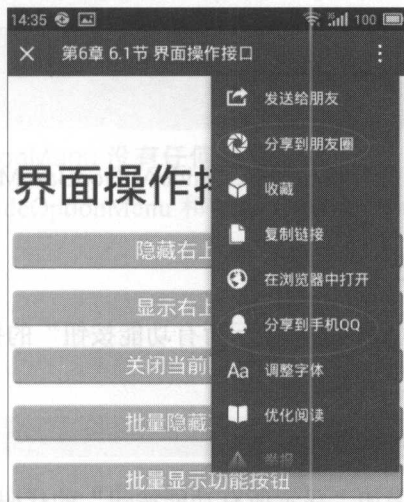


图 6.2 显示“分享到朋友圈”与“分享到 QQ”菜单

与上一节的 API 不同，当前 API 可以显示所有的菜单项，所有 menu 项请参考附录 C。

6.1.4 隐藏所有非基础按钮

在微信的 JSSDK 菜单体系中，可以分为“基本类”与“非基本类（传播类和保护类）”，本节的 API 主要是“隐藏所有非基础按钮”，名字叫“hideAllNonBaseMenuItem”，官方示例代码如下：

```
01 wx.hideAllNonBaseMenuItem();
```

在 sample.php 文件中，增加“隐藏所有非基础按钮”的代码结构如下：

```
01 <body>
02     <h1 style="font-size: 8em">:</h1>
03     <b style="font-size: 3em">界面操作接口！</b><br /><br />
04     <input type="button" value="隐藏所有非基础按钮" id="hideAllNonBaseMenuItem" /><br
05 /><br />
06 </body>
```

然后增加“隐藏所有非基础按钮”的响应事件，代码如下：

```
01 $("#hideAllNonBaseMenuItem").click(function(){ //批量隐藏
02     wx.hideAllNonBaseMenuItem({
03         hideAllNonBaseMenuItem:true
04     });
05 });
```

最后记得在 JSSDK 的配置环境中加入“hideAllNonBaseMenuItem”API 的权限。

6.1.5 显示所有功能按钮

最后一个与菜单有关的 API 是“显示所有功能按钮”的接口，名字是“showAllNonBaseMenuItem”，官方示例代码如下：

```
01 wx.showAllNonBaseMenuItem();
```

在 sample.php 文件中，增加“显示所有功能按钮”的代码结构如下：

```

01 <body>
02   <h1 style="font-size: 8em">.</h1>
03   <b style="font-size: 3em">界面操作接口! </b><br /><br />
04   <input type="button" value="显示所有功能按钮" id="showAllNonBaseMenuItem" /><br
05 /><br />
06 </body>

```

然后增加“显示所有功能按钮”的响应事件，代码如下：

```

01 //其他代码略

02   $("#showAllNonBaseMenuItem").click(function(){ //批量显示
03     wx.showAllNonBaseMenuItem({
04       showAllNonBaseMenuItem:true
05     });
06   });

```

最后记得在 JSSDK 的配置环境中加入“showAllNonBaseMenuItem”API 的权限。

6.2 常见问题

1. 有时候界面关闭不起作用，是怎么回事？

造成这个问题的因素主要有：JSSDK 环境问题、初始化是否完毕、设备是否卡顿、微信版本问题等。

2. 旧代码中的 hideOptionsMenu 没有任何效果？

新版本 JSSDK 取消了 hideOptionsMenu 和 showOptionsMenu 这两个操作，并且没有任何替代的方法。

第 7 章 JSSDK 的地理位置

在移动互联网时代，所有的操作都碎片化、移动化了，因此用户的需求更是不易把握。在一些数据测量分析工具中、在一些位置应用中（足迹、跑步类等），地理位置有着不可动摇的地位。尤其是随着 O2O、近场通信和移动互联网的发展，地理位置在微信中发挥着越来越重要的作用。本章就是用实战的形式演示微信中地理位置的应用。

7.1 地理位置

JSSDK 为开发者提供了两个位置 API 服务：获取地理位置和使用微信内置地图查看位置。

7.1.1 获取地理位置

第 1 个就是“获取地理位置”的 API，名为“getLocation”，官方示例代码如下：

```
01  wx.getLocation({
02      success: function (res) {          // 1 个参数，位置资源信息
03          var latitude = res.latitude;    // 纬度，浮点数，范围为 90 ~ -90
04          var longitude = res.longitude;  // 经度，浮点数，范围为 180 ~ -180。
05          var speed = res.speed;         // 速度，以米/每秒计
06          var accuracy = res.accuracy;   // 位置精度
07      }
08  });
```

在 sample.php 文件中，增加“获取地理位置”按钮，以及显示获取之后的位置信息，代码结构如下：


```

01      <style>
02          input{
03              width: 100%;
04              padding: 0.2em;
05              background-color: #5eb95e;
06              font-size: 1.4em;
07              background-image: linear-gradient(to bottom, #62c462, #57a957);
08              background-repeat: repeat-x;
09              color: #ffffff;
10              text-align: center;
11              text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
12              border-radius: 0.3em;
13          }
14          #info{
15              border-left: 3px solid #03a9f4;
16              background-color: #5eb95e;
17              color: #ffffff;
18              border-radius: 0.3em;
19              text-shadow: 0 -1px 0 rgba(0, 0, 0, 0.25);
20          }
21      </style>
22      <body>
23          <h1 style="font-size: 8em">.</h1>
24          <b style="font-size: 3em">位置操作接口！</b><br /><br />
25          <div id="info">
26              <p>纬度：<em id="latitude">无</em></p>
27              <p>经度：<em id="longitude">无</em></p>
28              <p>速度：<em id="speed">无</em></p>
29              <p>位置精度：<em id="accuracy">无</em></p>
30          </div>
31          <input type="button" value="获取当前地理位置信息" id="getLocation" />
32      </body>

```

然后在 wx.ready 中增加响应事件，代码如下：

```

01      wx.ready(function () {
02          var latitude, longitude, speed, accuracy;

```



```

03      $("#getLocation").click(function(){           //获取位置
04          wx.getLocation({
05              success:function (res) {
06                  latitude = res.latitude;             // 纬度，浮点数，范围为 90 ~ -90
07                  $("#latitude").html(latitude);
08                  longitude = res.longitude;           // 经度，浮点数，范围为 180 ~ -180。
09                  $("#longitude").html(longitude);
10                  speed = res.speed;                  // 速度，以米/每秒计
11                  $("#speed").html(speed);
12                  accuracy = res.accuracy;            // 位置精度
13                  $("#accuracy").html(accuracy);
14              });
15      });
16  });

```

最后记得在 JSSDK 的配置环境中加入“getLocation”API 的权限。

【代码解释】

“getLocation”的位置成功后，会返回“纬度”、“经度”、“速度”、“位置精度”的相关信息。当点击“获取位置”的按钮之后，微信会弹出提示信息，如图 7.1 所示。获取位置服务成功之后的结果如图 7.2 所示。

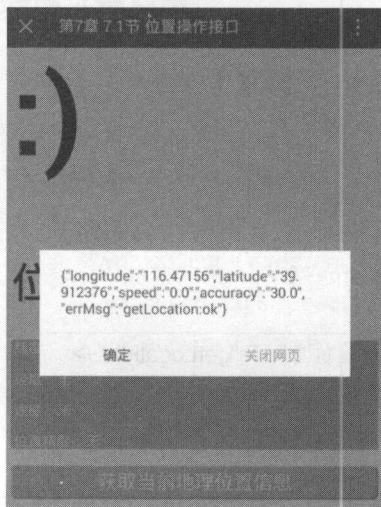


图 7.1 微信 JSSDK 获取位置信息提示

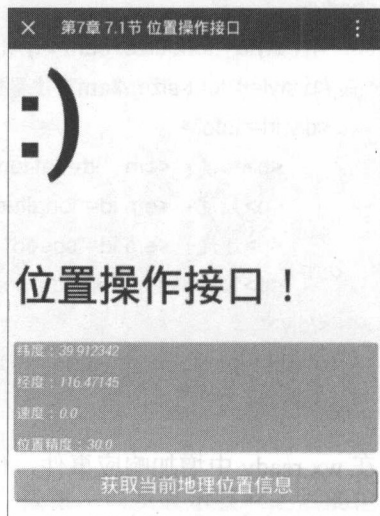


图 7.2 获取位置服务成功的信息

7.1.2 使用微信内置地图查看位置

只是有了地理位置数据信息还不够，应该要有视图方面的展示才有形象感、才比较靠谱。那么 JSSDK 的第 2 个位置服务 API 便是这个作用，名为“openLocation”，官方示例代码如下：

```
01 wx.openLocation({
02     latitude: 0,           // 纬度，浮点数，范围为 90 ~ -90
03     longitude: 0,         // 经度，浮点数，范围为 180 ~ -180。
04     name: "",             // 位置名
05     address: "",          // 地址详情说明
06     scale: 1,             // 地图缩放级别,整形值,范围从 1~28。默认为最大
07     infoUrl: ""           // 在查看位置界面底部显示的超链接,可点击跳转
08 });
```

在 sample.php 文件中，增加“使用微信内置地图查看位置”按钮，代码结构如下：

```
01 <body>
02     <h1 style="font-size: 8em">:</h1>
03     <b style="font-size: 3em">位置操作接口！</b><br /><br />
04     <div id="info">
05         <p>纬度：<em id="latitude">无</em></p>
06         <p>经度：<em id="longitude">无</em></p>
07         <p>速度：<em id="speed">无</em></p>
08         <p>位置精度：<em id="accuracy">无</em></p>
09     </div>
10     <input type="button" value="获取当前地理位置信息" id="getLocation" /><br /><br />
11     <input type="button" value="使用微信内置地图查看位置" id="openLocation" />
12 <br /><br />
13 />
</body>
```

然后增加响应事件，代码如下：

```
01 $("#openLocation").click(function(){ //地图查看
02     wx.openLocation({
03         latitude: latitude, // 纬度，浮点数，范围为 90 ~ -90
```

```
04         longitude: longitude, // 经度,浮点数,范围为 180 ~ -180
05         name: '测试',        // 位置名
06         address: '测试地址', // 地址详细说明
07         scale: 1,             // 地图缩放级别,整形值,范围从 1~28。默认为最大
08         infoUrl: 'http://www.html5waibao.com' // 在查看位置界面底部显示的超链接
09     });
10 }
```

最后记得在 JSSDK 的配置环境中加入“openLocation”API 的权限。

【代码解释】

开发者可以自定义位置服务 API 的相关信息,当调用位置服务 API 后,操作主动权便交给微信中的地图,如图 7.3 所示。

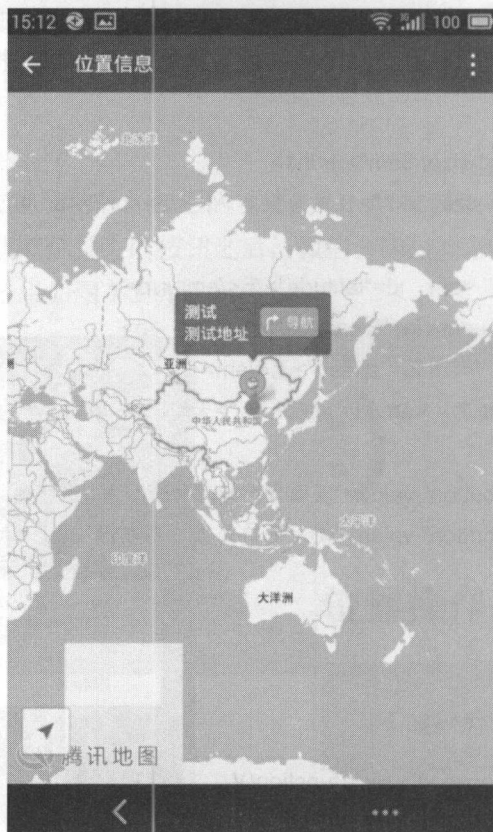


图 7.3 微信内的腾讯地图

7.2 常见问题

1. 为什么我的手机获取不到位置?

引起此问题的主要原因是手机设置问题,当弹出类似“此网页想获取您的位置信息”的提示时,需要点击“同意”按钮。如果还没法打开,苹果手机的设置是打开“设置”|“隐私”|“定位服务”,这里打开后,还需要在定位服务界面下找到微信,单击微信后选择“使用应用期间”。这样微信才能在苹果手机下使用定位服务。安卓手机比较简单,打开“设置”|“允许使用我的位置信息”即可。

第 8 章 JSSDK 的智能操作与微信支付

最近几年比较流行的创业技术方向中，智能设备是比较火的一种，只要和“智能”这两个字沾边的产品，基本上都融到资了。微信 JSSDK 也不甘落后，也在慢慢地布局自己的智能系统，目前只是开放了几个简单的 API，如智能接口、微信扫一扫、设备信息等。本章就主要来介绍这 3 个 API。

8.1 智能接口——识别音频并返回识别结果

在 JSSDK 中，为开发者提供了一个智能 API 服务。本节利用这个服务实现一个识别音频的应用。

在前几年，语音识别还是很高大上的技术，现在微信 JSSDK 开放了语音识别 API，做 JavaScript 开发的开发者就可以享受这种技术，而不必浪费精力从头研究了，降低了开发成本。

官方代码示例如下：

```
01 wx.translateVoice({
02     localId: "",           // 需要识别的音频的本地 ID，由录音相关接口获得
03     isShowProgressTips: 1, // 默认为 1，显示进度提示
04     success: function (res) {
05         alert(res.translateResult); // 语音识别的结果
06     }
07 });
```


由于识别语音需要语音的本地 ID，所以在识别语音前我们需要先录制一段语音，所以本例会用到前面学习过的语音 API：开始录音和结束录音。

在 sample.php 文件中，增加“录音”按钮，代码结构如下：

```
01 <body>
02     <h1 style="font-size: 8em">.</h1>
03     <b style="font-size: 3em">智能接口！</b><br /><br />
04     <div id="info">
05         <p>识别音频并返回识别结果接口：<em id="ai">请录音 . . . </em></p>
06     </div>
07     <input type="button" value="开始录音" id="startRecord" /><br /><br />
08     <input type="button" value="停止录音" id="stopRecord" /><br /><br />
09 </body>
```

然后在 wx.ready 中增加响应事件，代码如下：

```
01 wx.ready(function () {
02     var localId;
03     $("#startRecord").click(function(){ //开始录音
04         wx.startRecord();
05     });
06     $("#stopRecord").click(function(){ //停止录音
07         wx.stopRecord({
08             success:function(res){ //停止录音成功回调
09                 localId = res.localId;
10                 wx.translateVoice({
11                     localId: localId, // 需要识别的音频的本地 Id，由录音相关接口获得
12                     isShowProgressTips: 1, // 默认为 1，显示进度提示
13                     success: function (res) {
14                         $("#ai").html(res.translateResult); // 语音识别的结果
15                     }
16                 });
17             }
18         });
19     });
```



```
20    });
```

最后记得在 JSSDK 的配置环境中加入“translateVoice”、“startRecord”、“stopRecord”这 3 个 API 的权限。

【代码解释】

- 微信的智能接口主要是分析语义化，将音频转换成文本，第 14 行代码是将文本显示到页面中。
- 当点击“录音”按钮，录完声音之后，会显示微信分析的结果。

本例效果如图 8.1 所示。

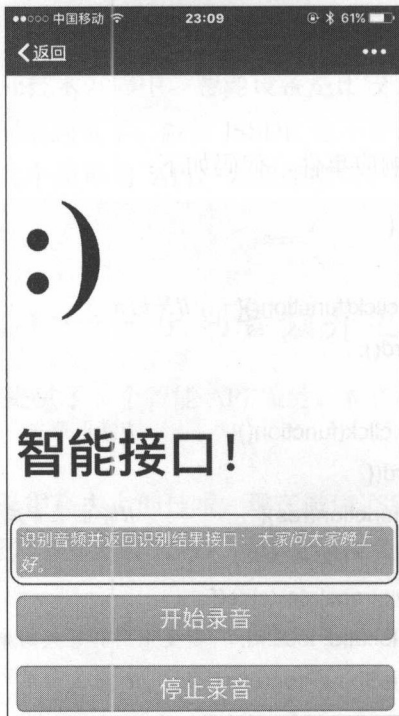


图 8.1 智能识别结果

8.2 微信扫一扫

除了之前的智能技术比较火之外，扫一扫也是扫遍全天下，其中“微信扫一扫”比其他软件的扫一扫精准度更高。本节就利用这个功能来实现一个微信扫一扫功能的应用。

微信扫一扫的调用相对简单，官方示例代码如下：

```
01 wx.scanQRCode({
02     needResult: 0, // 默认为 0，扫描结果由微信处理，1 则直接返回扫描结果，
03     scanType: ["qrCode", "barCode"], // 可以指定扫二维码还是一维码，默认二者都有
04     success: function (res) {
05         var result = res.resultStr; // 当 needResult 为 1 时，扫码返回的结果
06     }
07 });
```

在 sample.php 文件中，增加“扫一扫”按钮，代码结构如下：

```
01 <body>
02     <h1 style="font-size: 8em"></h1>
03     <b style="font-size: 3em"> 微信扫一扫接口! </b><br /><br />
04     <input type="button" value="点击扫一扫" id="scanQRCode" /><br /><br />
05 </body>
```

然后在 wx.ready 中增加响应事件，代码如下：

```
01 wx.ready(function () {
02     $("#scanQRCode").click(function() { //扫一扫
03         wx.scanQRCode({
04             needResult: 0, // 默认为 0，扫描结果由微信处理，1 则直接返回扫描结果，
05             scanType: ["qrCode", "barCode"], // 可以指定扫二维码还是一维码，默认二者都有
06             success: function (res) {
07                 var result = res.resultStr; // 当 needResult 为 1 时，扫码返回的结果
08                 alert(result);
09             }
10         });
11     });
12 });
```

最后记得在 JSSDK 的配置环境中加入“scanQRCode”API 的权限。

【代码解释】

- 微信的扫一扫 API 主要是做一维码与二维码分析。

- 扫码之后，会以字符串的形式返回扫码结果。
- 如扫描的是公众号的二维码，则会出现是否“关注”的信息，和微信直接调用“扫一扫”功能一样。

8.3 设备信息——获取网络状态接口

设备信息对于程序的伸缩性、适应性及运维、产品分析都有很大的价值，例如对微信 JSSDK 提供的网络状态分析，前端开发者可以按照不同的网络状态加载不同的样式、图片，运维可以了解用户的分布，产品分析可以知道用户的体验状态。

目前微信 JSSDK 提供了一个 API，可以让开发者获取用户设备的网络状态，名为“getNetworkType”，官方示例代码如下：

```
01  wx.getNetworkType({
02      success: function (res) {
03          var networkType = res.networkType; // 返回网络类型 2G, 3G, 4G, WiFi
04      }
05  });
```

在 sample.php 文件中，增加“点击获取网络状态”按钮，代码结构如下：

```
01  <body>
02      <h1 style="font-size: 8em">.</h1>
03      <b style="font-size: 3em"> 获取网络状态接口！ </b><br /><br />
04      <div id="info">
05          <p>网络状态：<em id="networkType">等待获取 . . . </em></p>
06      </div>
07      <input type="button" value="点击获取网络状态" id="getNetworkType" /><br /><br />
08  />
</body>
```

然后在 wx.ready 中增加响应事件，代码如下：

```
01  wx.ready(function () {
02      $("#getNetworkType").click(function(){ //获取网络状态
03          wx.getNetworkType({
```

```

04      success: function (res) {
05          var networkType = res.networkType;    // 返回网络类型 2g、3g、4g、wifi
06          $("#networkType").html(networkType);
07      }
08  });
09  });
10  });

```

最后记得在 JSSDK 的配置环境中加入 “getNetworkType” API 的权限。

【代码解释】

微信的网络状态参数有 4 种，包含 2G、3G、4G、WiFi，本例效果如图 8.2 所示。

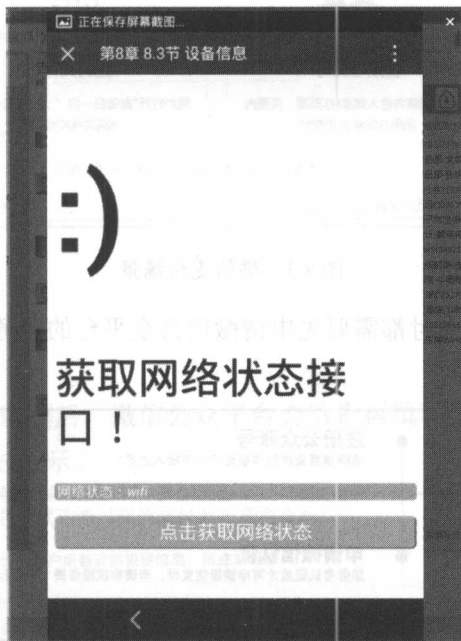


图 8.2 微信 JSSDK 获取网络状态 WiFi

8.4 微信支付

目前微信支持网页内完成支付，主要流程是：

(1) 用户打开商户网页选购商品，发起支付，在网页通过 JavaScript 调用

getBrandWCPayRequest 接口，发起微信支付请求，用户进入支付流程。

(2) 用户成功支付并点击完成按钮后，商户的前端会收到 JavaScript 的返回值。商户可直接跳转到支付成功的静态页面进行展示。

(3) 商户后台收到来自微信开放平台的支付成功回调通知，标志该笔订单支付成功。

8.4.1 微信支付公众平台申请流程

微信支付有四种应用场景，如图 8.3 所示。



图 8.3 微信支付场景

除 APP 支付外，其他支付都需要先申请微信公众平台的服务号并通过认证。认证流程如图 8.4 所示。

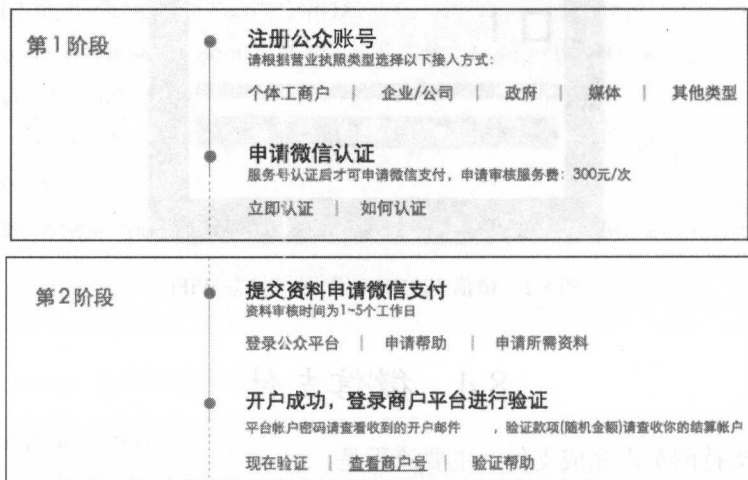


图 8.4 微信支付认证流程

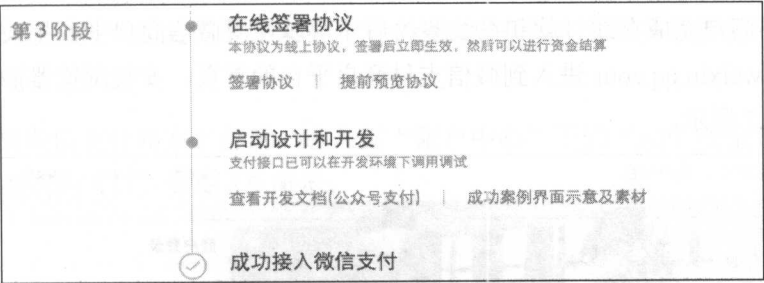


图 8.4 微信支付认证流程（续）

微信支付涉及资金往来，所以需要在认证的时候提交结算账户，如图 8.5 所示。

账户类型：对公账户 企业类型帐号，结算账户不支持修改私人账户。

开户名称：深圳市腾讯计算机系统有限公司
若【个体工商户】没有对公账户，请选择法人账户。如没有法人的选项，请返回

*开户银行：为【个体工商户】；如拉取的法人姓名不正确，请返回上一步【商户信息】，将
 找不到所在的开户银行？

*开户银行城市：
如果找不到所在城市，请选择所在地区或者上级城市

*开户支行： 找不到所在的支行？

图 8.5 结算账户

8.4.2 完成账户验证

提交资料完成并审核通过后，微信公众平台会给资料填写中的重要邮箱发送微信支付商户平台的信息，如图 8.6 所示。

恭喜你，成功通过微信支付商户资料审核！

以下为你的账户所需要的重要信息，请注意保密：

微信支付商户号	<input type="text"/>
商户平台登录帐号	<input type="text"/>
商户平台登录密码	<input type="text"/>
申请对应的公众号	<input type="text"/>
公众号APPID	<input type="text"/>

图 8.6 商户平台账号

此时需要商户完成在线打款和在线签约后才可以使⤵用微信商户平台相关功能。首先访问 <https://pay.weixin.qq.com> 进入到微信支付商户平台的首页，安装浏览器插件后即可登录使用，如图 8.7 所示。

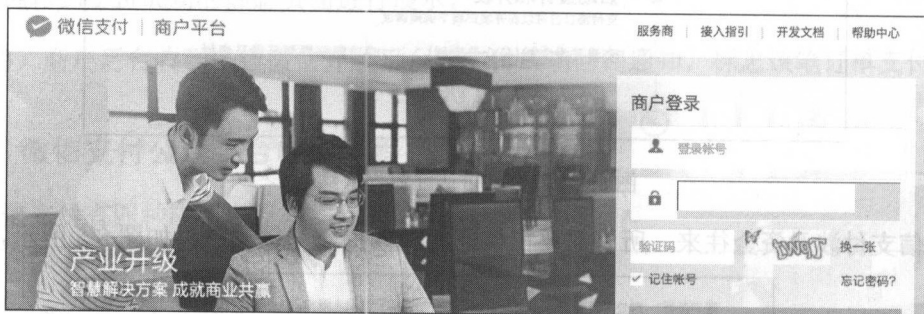


图 8.7 登录商户平台

登录成功后在“账户中心”中找到“账户概览”进行账户验证并签署协议，如图 8.8 所示。

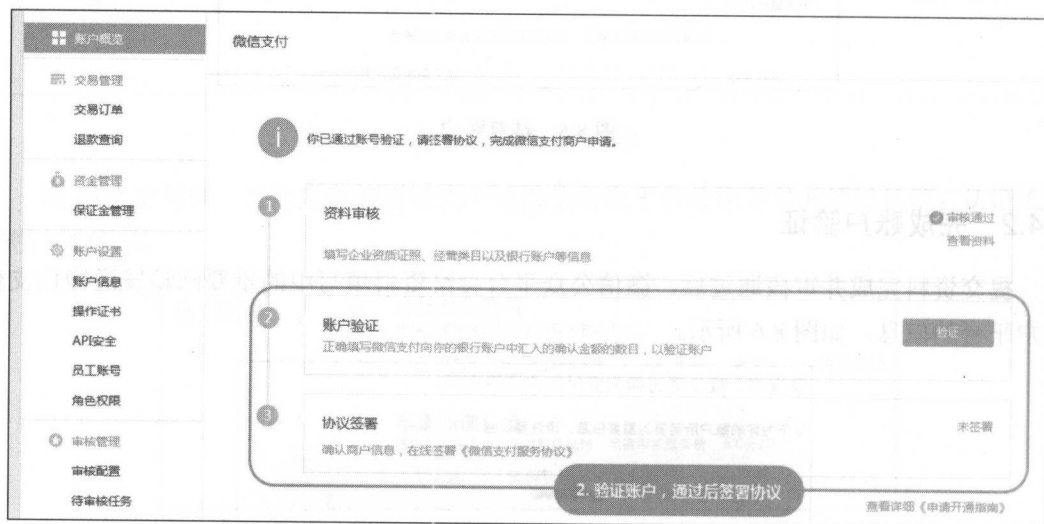


图 8.8 完成账户验证和协议签署

完成以上申请接入步骤后，微信支付的申请工作就已经完成了。

8.4.3 设置 API 秘钥

因为微信支付涉及资金往来，所以安全很重要，为了防止别人恶意篡改数据，API 接

口在调用的时候需要按照指定的规则来对开发者请求进行签名，服务器接到后会对签名做验证。

用户登录到微信支付商家平台后，进入到“账户中心”下的“API 安全”界面，可以找到设置 API 密钥的入口，如图 8.9 所示。

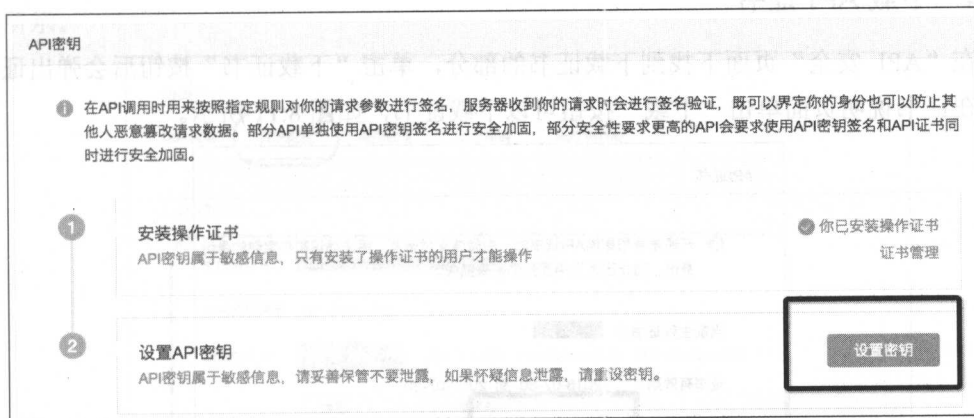


图 8.9 设置 API 密钥

单击“设置密钥”按钮后效果如图 8.10 所示。

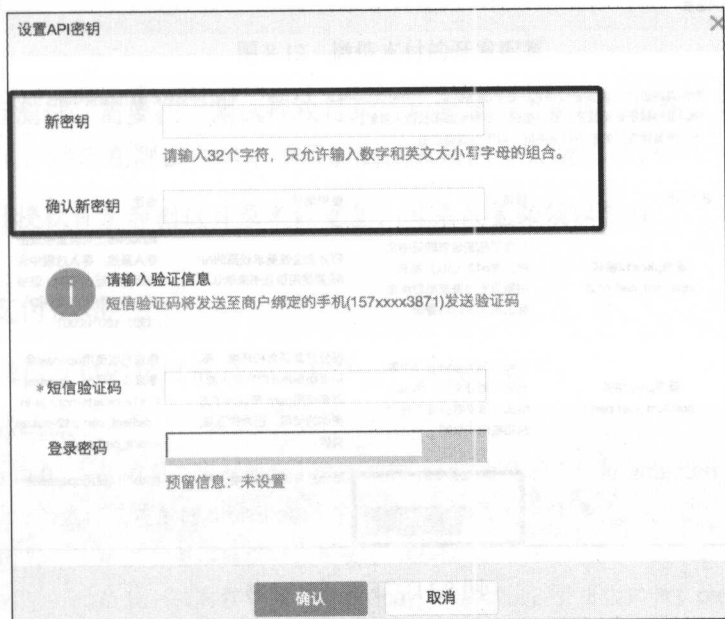


图 8.10 设置 API 的新密钥

注意：密钥的格式为 32 位长度的英文数字组合，可以有大小写。这个密钥可以借助 PHP 函数 md5() 来辅助生成。用户单击“确认”按钮即可设置完成 API 密钥。这个密钥后面会使用，所以一定要登记好。

8.4.4 下载 API 证书

在“API 安全”页面下找到下载证书的部分，单击“下载证书”按钮后会弹出证书说明，在证书说明页面单击“下载”按钮可以下载证书，如图 8.11 所示。

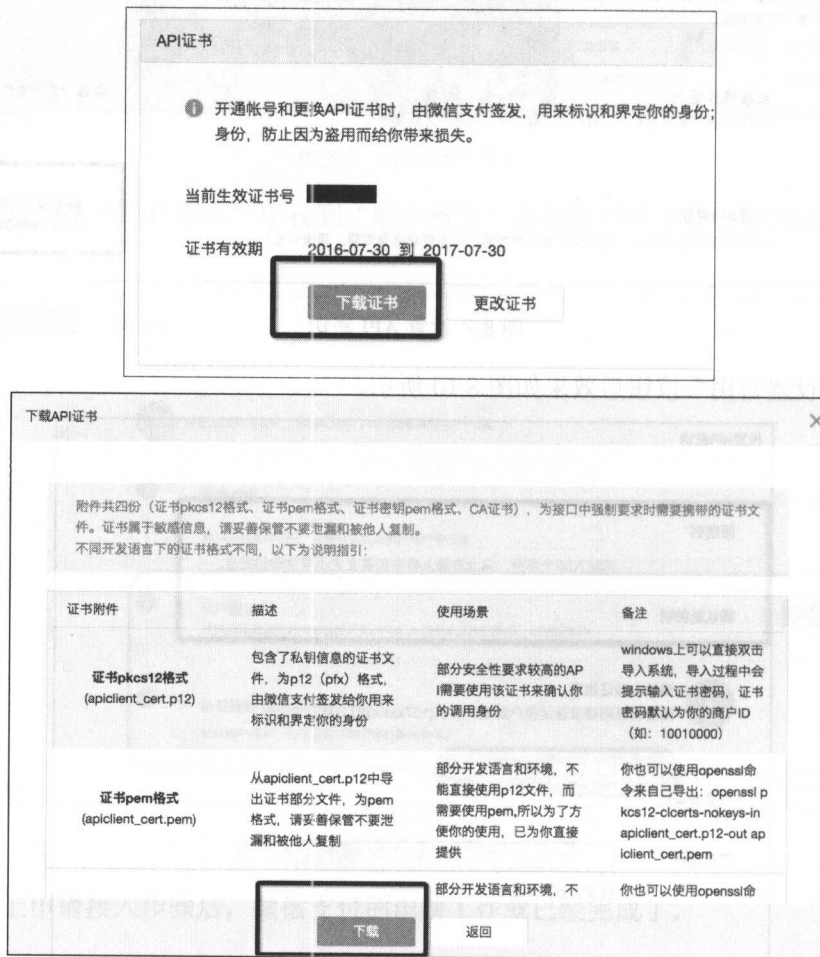


图 8.11 下载 API 证书

用户下载成功后解压 zip 压缩包即可获取和使用 API 证书。

注意：下载证书前需要先进行 API 密钥的设置，每次下载都会进行手机号验证。

8.4.5 微信支付的开发配置

进入微信公众号后台，打开“微信支付”界面，然后选择第 2 项“开发配置”，如图 8.12 所示。

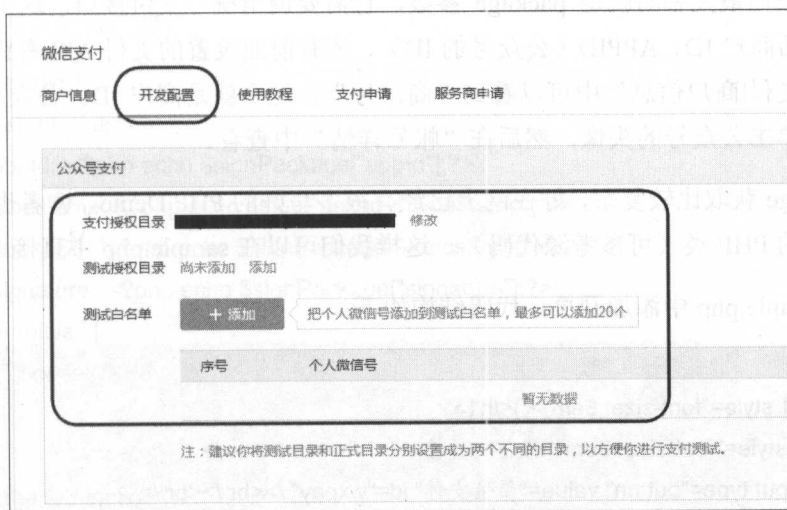


图 8.12 微信支付的开发配置

如果是进行测试，需要在“测试授权目录”后面单击“添加”按钮，添加我们之前设置好的测试 URL，然后在测试白名单中添加测试人员的个人微信号。

注意：支付授权目录和测试目录不能重复。目录设置必须以斜杠“/”结尾。

8.4.6 微信支付测试

JSSDK 中微信支付的接口是 `chooseWXPAY`，其代码如下：

```
wx.chooseWXPAY({
    timestamp: 0, // 支付签名时间戳，注意微信 jssdk 中的所有使用 timestamp 字段均为小写。
    // 但最新版的支付后台生成签名使用的 timeStamp 字段名需大写其中的 S 字符
    nonceStr: "", // 支付签名随机串，不长于 32 位
    package: "", // 统一支付接口返回的 prepay_id 参数值，提交格式如：prepay_id=***
    signType: "", // 签名方式，默认为'SHA1'，使用新版支付需传入'MD5'
```

```

    paySign: "",      // 支付签名
    success: function (res) {
        // 支付成功后的回调函数
    }
});

```

这里我们最关心的就是 `package` 参数，它需要调用统一支付接口，这个接口需要很多参数，包括商户 ID、APPID（公众号的 ID）、还有前面设置的支付 API 密钥。在公众号后台“微信支付|商户信息”中可以看到“商户号”，这个就是商户 ID。公众号 ID 可以在登录后后台单击公众号的头像，然后在“帐号详情”中查看。

Package 获取比较复杂，好在网上已经有很多写好的 PHP Demo，读者也可以直接下载一个写好的 PHP 类（可参考源代码），这样我们可以在 `sample.php` 中直接调用。

在 `sample.php` 中添加网页，代码结构如下：

```

<body>
    <h1 style="font-size: 8em">:</h1>
    <b style="font-size: 3em">微信支付接口！ </b><br /><br />
    <input type="button" value="微信支付" id="wxpay" /><br /><br />
</body>

```

然后添加 PHP 代码，获取需要的参数：

```

01  $wechatAppPay      =      new      wxpay("gh_3ed204d47642",      "1274579601",
"http://1.weixinjssdk6.applinzi.com/", "C380BEC2BFD727A4B6845133519F3AD6 ");
02  $params['body'] = '商品描述';                                //商品描述
03  $params['out_trade_no'] = 'O20160617021323-001';            //自定义的订单号
04  $params['total_fee'] = '100';                                //订单金额 只能为整数 单位为分
05  $params['trade_type'] = 'JSAPI';                             //交易类型 JSAPI | NATIVE | APP | WAP
06  $result = $wechatAppPay->unifiedOrder( $params );
07
08  $prepay_id = "prepay_id=" . @$wechatAppPay->getAppPayParams( $result['prepayid'] );
09  $noncestr=$result['noncestr'];
10  $timestamp=$result['timestamp'];
11  $sign=$result['sign'];

```


【代码解释】

- 第1行调用已经设计好的 PHP 支付类 wxpay。
- 第6行获取包含所有参数的变量 result。
- 第8~11行设置接口需要的各个变量，注意 package 变量需要以“prepay_id=”这种形式开始。

最后添加微信配置信息：

```

01  wx.config({
02      debug: true,
03      appId: '<?php echo $signPackage["appId"];?>',
04      timestamp: '<?php echo $signPackage["timestamp"];?>',
05      nonceStr: '<?php echo $signPackage["nonceStr"];?>',
06      signature: '<?php echo $signPackage["signature"];?>',
07      jsApiList: [
08          "chooseWXPay"
09      ]
10  });
11  wx.ready(function () {
12      $("#wxpay").click(function(){
13          wx.chooseWXPay({
14              timestamp: '<?php echo $timestamp;?>', // 支付签名时间戳
15              nonceStr: '<?php echo $noncestr;?>', // 支付签名随机串
16              package: '<?php echo $prepay_id;?>', // 统一支付接口返回的 prepay_id 参数值
17              signType: 'MD5', // 签名方式 MD5
18              paySign: '<?php echo $sign;?>', // 支付签名
19              success: function (res) {
20                  if(res.err_msg === 'get_brand_wcpay_request:ok'){
21                      alert('支付成功，返回订单列表！');
22                  }else if(res.err_msg === 'get_brand_wcpay_request:cancel'){
23                      alert('取消支付！');
24                  }
25              }
26          });
27      });

```


28 });

好了，更新服务器上的网页，微信支付效果如图 8.13 所示。

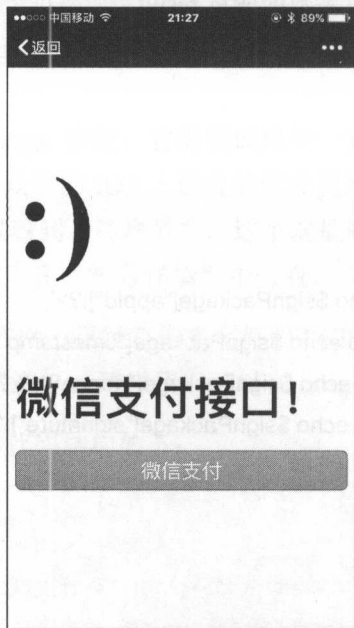


图 8.13 微信支付效果

8.5 常见问题

1. 为什么微信中的扫一扫功能无法执行？

首先检查摄像头是否可用，然后检查摄像头是不是设置了拒绝使用，最后看看所安装的微信的版本。

2. 为什么我的录音解析文本不正确？

语义音频分析也是一个非常大的课题，里面有很多分析算法，有一定的误差或者不准确属于正常范围。

3. 微信 JSSDK 的微信支付和 JSAPI 的微信支付调用有区别吗？

首先两者都能实现微信支付，可能是微信为保留两种开发状态而设置的。

JSSDK 的微信支付使用的是 `wx.chooseWXPay`，而 JSAPI 使用的是 `WeixinJSBridge`。

两者都只是在微信内部浏览器中使用，两者的参数类似，获取参数的方法也类似。

WeixinJSBridge 的调用方式如下：

```
function onBridgeReady(){
    WeixinJSBridge.invoke(
        'getBrandWCPayRequest', {
            "appId" : "wx2421b1c4370ec43b",    //公众号名称，由商户传入
            "timeStamp": "1395712654",        //时间戳，自 1970 年以来的秒数
            "nonceStr" : "e61463f8efa94090b1f366cccfbbb444", //随机串
            "package" : "prepay_id=u802345jgjsdfgsdg888",
            "signType" : "MD5",                //微信签名方式
            "paySign" : "70EA570631E4BB79628FBCA90534C63FF7FADD89" //微信签名
        },
        function(res){
            if(res.err_msg == "get_brand_wcpay_request: ok" ) {} // 使用以上方式判断前端返回,微信团队郑重提示：res.err_msg 将在用户支付成功后返回 ok，但并不保证它绝对可靠。
        }
    );
}

if (typeof WeixinJSBridge == "undefined"){
    if( document.addEventListener ){
        document.addEventListener('WeixinJSBridgeReady', onBridgeReady, false);
    }else if (document.attachEvent){
        document.attachEvent('WeixinJSBridgeReady', onBridgeReady);
        document.attachEvent('onWeixinJSBridgeReady', onBridgeReady);
    }
}else{
    onBridgeReady();
}
```



第 9 章 JSSDK 的微信商店

微信一直在加速对企业的支持，其中对微商业的生态支持比较全面，例如本章介绍的微信商店的接口。所谓微商，最开始指用手机来开店卖货，渐渐地，微商成了在微信里开店的商家。如果读者掌握了本章的功能，微信开店就大为方便了。

9.1 微信小店

微信小店基于微信支付，包括添加商品、商品管理、订单管理、货架管理、维权等功能。开发者可以通过小店接口来实现快速地开店。本节通过实例来学习这个接口的使用。

微信商店的主要业务活动就是买卖商品，所以微信提供的接口便是“跳转至微信商品页”，名字叫“openProductSpecificView”，官方示例代码如下：

```
01  wx.openProductSpecificView({  
02      productId: "", // 商品 id  
03      viewType: " // 0.默认值，普通商品详情页 1.扫一扫商品详情页 2.小店商品详情页  
04  });
```

在 sample.php 文件中，增加“跳转微信商品页”按钮，代码结构如下：

```
01  <body>  
02      <h1 style="font-size: 8em">.</h1>  
03      <b style="font-size: 3em">微信小店接口! </b><br /><br />  
04      <input type="button" value="跳转微信商品页" id="openProductSpecificView" />  
05  <br /><br />  
    </body>
```

然后在 wx.ready 中增加响应事件，代码如下：

```
01 wx.ready(function () {
02     $("#openProductSpecificView").click(function(){//获取网络状态
03         wx.openProductSpecificView({
04             productId: 'pDF3iY_m2M7EQ5EKKKWd95kAxfNw', // 商品 id
05             viewType: " // 0.默认值，普通商品详情页 1.扫一扫商品详情页 2.小店商品详情页
06         });
07     });
08 });
```

最后记得在 JSSDK 的配置环境中加入 “openProductSpecificView” API 的权限。

【代码解释】

- 微信的跳转微信商品页接口中，主要需要注意的是“商品 id”。
- 接口的第 2 个参数主要是商品的来源类型。

9.2 微信卡券

除了正常的商品售卖之外，商店经营的过程中，还需要进行一些必要的促销、让利活动，这时候就会用到“微信卡券”接口。

9.2.1 批量添加卡券

第 1 个要介绍的是“批量添加卡券”接口，名字叫“addCard”，官方示例代码如下：

```
01 wx.addCard({
02     cardList: [{
03         cardId: "",
04         cardExt: ""
05     }], // 需要添加的卡券列表
06     success: function (res) {
07         var cardList = res.cardList; // 添加的卡券列表信息
08     }
09 });
```


在 sample.php 文件中, 增加“批量添加卡券”按钮, 代码结构如下:

```
01 <body>
02     <h1 style="font-size: 8em">.</h1>
03     <b style="font-size: 3em">微信卡券接口! </b><br /><br />
04     <input type="button" value="批量添加卡券接口" id="addCard" /><br /><br />
05 </body>
```

然后在 wx.ready 中增加响应事件, 代码如下:

```
01 wx.ready(function () {
02     $('#addCard').click(function(){//添加卡券
03         wx.addCard({
04             cardList: [
05                 {
06                     cardId: 'pDF3iY9tv9zCGCj4jTXFOo1DxHdo',
07                     cardExt: '{"code": "", "openid": "", "timestamp": '+Math.round(new
Date().getTime()/1000)+'', "signature":'+wx.config.signature+'}'
08                 },
09                 {
10                     cardId: 'pDF3iY9tv9zCGCj4jTXFOo1DxHdo',
11                     cardExt: '{"code": "", "openid": "", "timestamp": '+Math.round(new
Date().getTime()/1000)+'', "signature":'+wx.config.signature+'}'
12                 }
13             ],
14             success: function (res) {
15                 var cardList = res.cardList; // 添加的卡券列表信息
16                 alert('已添加卡券: ' + JSON.stringify(cardList));
17             }
18         });
19     });
});
```

最后记得在 JSSDK 的配置环境中加入“addCard”API 的权限。

【代码解释】

微信的批量添加卡券接口中，需要注意卡券列表的有效性。

9.2.2 调用适用于门店的卡券列表并获取用户选择列表

微信卡券第2个相关的接口是“调用适用于门店的卡券列表并获取用户选择列表”，名称叫“chooseCard”官方示例代码如下：

```

01  wx.chooseCard({
02      shopId: "",          // 门店 Id
03      cardType: "",        // 卡券类型
04      cardId: "",          // 卡券 Id
05      timestamp: 0,        // 卡券签名时间戳
06      nonceStr: "",        // 卡券签名随机串
07      signType: "",        // 签名方式，默认'SHA1'
08      cardSign: "",        // 卡券签名，详见附录 D
09      success: function (res) {
10          var cardList= res.cardList; // 用户选中的卡券列表信息
11      }
12  });

```

在 sample.php 文件中，增加“调用适用于门店的卡券列表并获取用户选择列表”按钮，代码结构如下：

```

01  <body>
02      <h1 style="font-size: 8em">.</h1>
03      <b style="font-size: 3em">微信卡券接口！ </b><br /><br />
04      <input type="button" value="批量添加卡券接口" id="addCard" /><br /><br />
05      <input type="button" value="调用适用于门店的卡券列表并获取用户选择列表"
06      id="chooseCard" /><br /><br />
07  </body>

```

然后在 wx.ready 中增加响应事件，代码如下：

```

01      $("#chooseCard").click(function(){ //调用卡券
02          wx.chooseCard({
03              cardSign: '97e9c5e58aab3bdf6fd6150e599d7e5806e5cb91',

```



```

04         timestamp: Math.round(new Date().getTime()/1000),
05         nonceStr: 'k0hGdSXkZEj3Min5',
06         success: function (res) {
07             var cardList= res.cardList; // 用户选中的卡券列表信息
08             alert('已选择卡券: ' + JSON.stringify(cardList));
09         }
10     });
11 });

```

最后记得在 JSSDK 的配置环境中加入“chooseCard”API 的权限。

【代码解释】

调用适用于门店的卡券列表并获取用户选择列表接口中，需要注意卡券签名、卡券 id。

9.2.3 查看微信卡包中的卡券

第 3 个接口是查看微信卡包中的卡券，名称叫“openCard”，官方示例代码如下：

```

01 wx.openCard({
02     cardList: [{
03         cardId: "",
04         code: ""
05     }]/ 需要打开的卡券列表
06 });

```

在 sample.php 文件中，增加“查看微信卡包中的卡券”按钮，代码如下：

```
<input type="button" value="查看微信卡包中的卡券" id="openCard" /><br /><br />
```

然后在 wx.ready 中增加“查看微信卡包中的卡券”响应事件按钮，代码如下：

```

01 //其他代码略
alert('只是测试之用···您没有当前公众号的卡券，不能打开卡券。');
02 $("#openCard").click(function(){ //调用卡券
03     alert('只是测试之用···您没有当前公众号的卡券，不能打开卡券。');
04     wx.openCard({
05         cardList: [{
06             cardId: "",

```

```
07         code: "  
08         }// 需要打开的卡券列表  
09     });  
10 };  
11 };
```

最后记得在 JSSDK 的配置环境中加入“openCard”API 的权限。

【代码解释】

由于申请的公众号不提供真实的微店环境，没有卡券列表，所以本章的代码仅供参考。在实际的开发中，请以真实的 cardList 列表配置。

9.3 常见问题

1. 卡券签名 (cardSign) 怎么获取?

如果您会获取“api_ticket”，那么“cardSign”也是类似的算法。将“api_ticketapp_id”、“location_id”、“times_tamp”、“nonce_str”、“card_id”和“card_type”的 value 值进行字符串的字典序排序，然后把所有参数字符串拼接成一个字符串进行 sha1 加密，就会得到“cardSign”。

第 10 章 微信网页开发样式库

视觉体验已经是当今 APP 推广的重要一环，为了更好地开发微信网页和小程序，微信推出了一套网页开发样式库，统称为 WeUI，它是一套同微信原生视觉体验一致的基础样式库。本章将学习 WeUI 的优势、WeUI 的使用。

10.1 认识 WeUI

WeUI 目前包含表单、基础组件、操作反馈、导航相关、搜索相关、层级规范等内容，已经在 GitHub 上开源，读者可以访问 <http://weui.io> 进行预览。

10.1.1 WeUI 的特色

微信的 WeUI 不仅仅在网页上支持，还可以用于小程序的开发。它主要具备如下优势：

- 同微信客户端一致的视觉效果，令所有微信用户都能更容易地使用你的网站或小程序。
- 便捷获取、快速使用，降低开发和设计成本。
- 微信设计团队精心打造，样式清晰明确、简洁大方。

WeUI 的样式如图 10.1 所示，可以看到其中包含了 Button、Cell、Dialog、Progress、Toast、Article、ActionSheet、Icon 等微信中常见的各式元素。

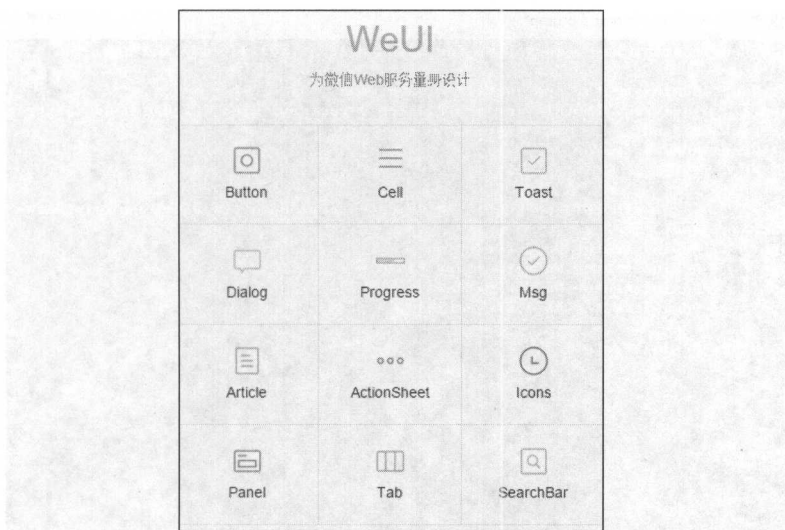


图 10.1 WeUI 基本样式

10.1.2 安装 WeUI

微信官方提供两种方式获取 WeUI: bower 包和 npm 包。这和传统的 msi 安装方式不同,读者一定要注意。两者的安装命令相似,分别是:

<code>bower install --save weui</code>	#bower 包安装方式
<code>npm install --save weui</code>	#npm 包安装方式

如果读者已经安装了 Node.js,则估计对 npm 包已经很熟悉了,本书以 npm 包的安装方式进行详细讲解。

(1) npm 是 Node.js 默认的包管理方式,我们可以先在系统中安装 Node.js,下载地址是 <https://nodejs.org/en/>。下载后是 msi 格式,可以直接安装,具体过程这里就不说了。安装后会有一项“Node.js command prompt”,这是它的命令行窗口,打开该窗口。

(2) 在命令行窗口中输入:

<code>npm install --save weui</code>	#npm 包安装方式
--------------------------------------	------------

安装效果如图 10.2 所示。

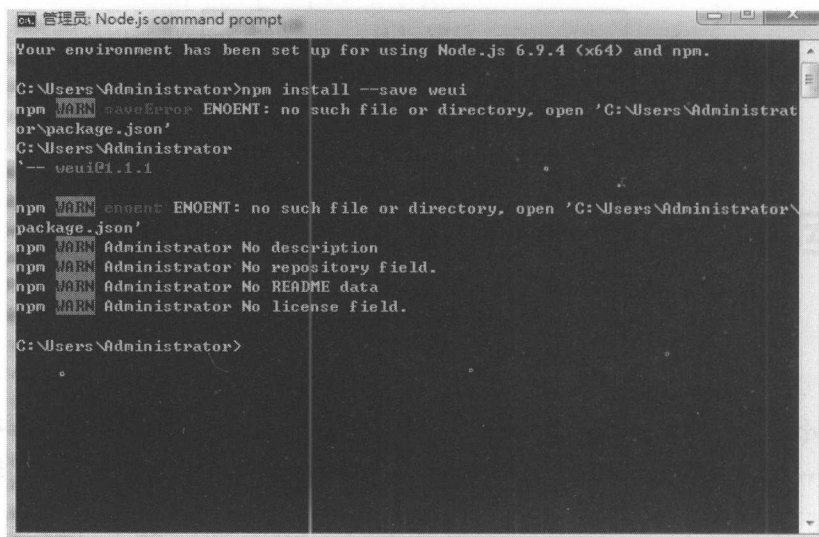


图 10.2 WeUI 安装效果

(3) 默认安装位置是在当前执行命令的文件夹下:

C:\Users\Administrator\node_modules\weui

其中 node_modules 是 Node.js 安装模块的默认文件夹, 如果不想安装在 C 盘, 也可以在命令行窗口下使用 cd 命令切换文件夹。

(4) 打开 weui 文件夹, 其结构如图 10.3 所示。

名称	修改日期	类型	大小
dist	2017/1/26 星期...	文件夹	
src	2017/1/26 星期...	文件夹	
.npmignore	2017/1/26 星期...	NPMIGNORE 文件	1 KB
.travis.yml	2017/1/26 星期...	YML 文件	1 KB
bower.json	2017/1/26 星期...	JSON 文件	1 KB
CHANGELOG.md	2017/1/26 星期...	MD 文件	12 KB
CONTRIBUTING.md	2017/1/26 星期...	MD 文件	2 KB
gulpfile.js	2017/1/26 星期...	JScript Script 文件	5 KB
package.json	2017/1/26 星期...	JSON 文件	3 KB
README.md	2017/1/26 星期...	MD 文件	2 KB

图 10.3 WeUI 结构

其中的 dist 文件夹下有两个文件夹: example 和 style。example 是 WeUI 的使用示例, style 是 WeUI 的样式库。

(5) 打开 example 文件夹下的 index.html 文件, 如图 10.4 所示, 这是所有组件的演示示例。



图 10.4 组件的演示示例

注意: 这样就获取了微信的 WeUI 整体库。如果只是使用微信的样式, 其实我们复制一个 weui.css 放在自己的项目中即可。

10.1.3 使用 WeUI 做一个欢迎界面

在安装完成的 WebUI 里面有个 style 文件夹, 其中有两个样式文件: weui.min.css 和 weui.css。两个样式的内容基本一致, 常使用各种框架的读者应该知道, 调试时使用 weui.css, 可以方便地查看代码, 而正式交付时建议使用 weui.min.css, 体积更小, 加载更迅速。

我们使用 WeUI 的重点就是使用 weui.css 中的样式。下面设计一个 HTML 5 页面 welcome.html, 保存在 dist 文件夹下。

```
01 <!DOCTYPE html>
02 <html>
03 <head>
```



```

04      <meta charset="UTF-8">
05      <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=0">
06      <title>WeUI</title>
07      <link rel="stylesheet" href="./style/weui.css"/>
08      </head>
09      <body>
10      <a href="javascript:;" class="weui-btn weui-btn_mini weui-btn_primary">欢迎</a>
11      </body>
12      </html>

```

【代码解释】

- 第 7 行引入 WeUI 的样式库 weui.css。
- 第 10 行设计一个按钮，通过 class 属性使用了组合样式。微信样式库示例中给出的按钮都是使用了<a>实现，其实也可以使用<input>，但建议都是<a>。

本例效果就是一个应用了简单样式的按钮，如图 10.5 所示。

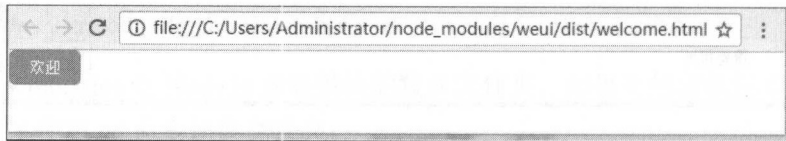


图 10.5 应用了 WeUI 的按钮

10.2 使用表单

网页样式库支持的表单有 Button、Input、List、Slider、Uploader。本节分别介绍它们的使用情况。

10.2.1 按钮（Button）

微信中按钮的样式如图 10.6 所示，这些网页样式库都支持。

在网页提供的样式中，所有按钮必需使用基本样式 weui_btn。除此以外，按钮主要分为两种样式：操作类样式和效果类样式。图 10.6 上方的都是操作类样式，如警告类操作；图 10.6 下方都是效果类样式，如按钮的 mini 样式。



图 10.6 按钮样式

下面我们分别使用各种样式来创建一个类似图 10.6 的按钮图，读者可以看到每个按钮样式的效果。

```

01  <!DOCTYPE html>
02  <html>
03  <head>
04      <meta charset="UTF-8">
05      <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=0">
06  <title>按钮</title>
07  <link rel="stylesheet" href="style/weui.css"/>
08  </head>
09  <body>
10  <p>操作按钮：</p>
11  <a href="javascript:;" class="weui-btn weui-btn_primary">按钮</a>
12  <a href="javascript:;" class="weui-btn weui-btn_disabled weui-btn_primary">按钮置灰</a>
13  <a href="javascript:;" class="weui-btn weui-btn_warn">确认</a>
14  <a href="javascript:;" class="weui-btn weui-btn_disabled weui-btn_warn">确认置灰</a>

```

```

15 <a href="javascript:;" class="weui-btn weui-btn_default">按钮</a>
16 <a href="javascript:;" class="weui-btn weui-btn_disabled weui-btn_default">按钮置灰</a>
17 <p>效果按钮: </p>
18 <a href="javascript:;" class="weui-btn weui-btn_plain-default">按钮</a>
19 <a href="javascript:;" class="weui-btn weui-btn_plain-primary">按钮</a>
20 <a href="javascript:;" class="weui-btn weui-btn_mini weui-btn_primary">按钮</a>
21 <a href="javascript:;" class="weui-btn weui-btn_mini weui-btn_default">按钮</a>
22 </body>
23 </html>

```

【代码解释】

- 第 7 行首先引入微信的网页样式库。
- 第 11~21 行设计了 10 个按钮，并使用不同的组合样式。一定要注意，每个组合样式的第 1 个样式是 weui-btn。
- 一定要注意样式的名称，不是统一的下画线“_”，也不是统一的中画线“-”，这里在命名时有一点乱，主样式是“-”，辅样式是“_”。

本例按钮效果如图 10.7 所示。

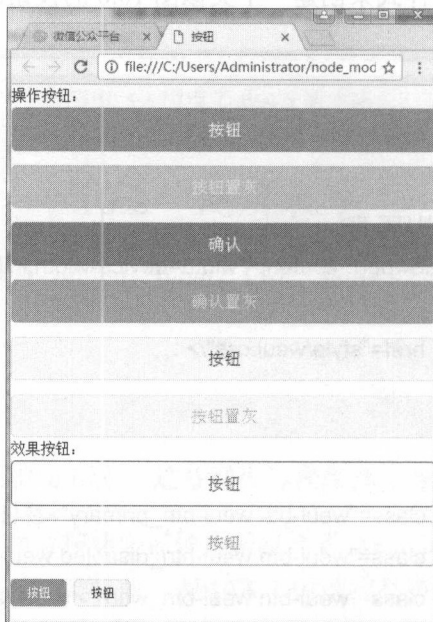


图 10.7 按钮效果

10.2.2 表单输入 (Input)

微信中的表单输入和 HTML 的表单输入类似, 包含了单选列表、复选列表、日期输入、开关、文本框、文本域等。下面通过一个注册界面来学习如何使用表单输入的样式。新建一个 `signup.html` 文件, 输入如下代码:

```
01  <!DOCTYPE html>
02  <html>
03  <head>
04      <meta charset="UTF-8">
05      <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=0">
06  <title>输入表单</title>
07  <link rel="stylesheet" href="style/weui.css"/>
08  </head>
09  <body>
10      <div class="weui-cells__title">姓名</div>
11      <div class="weui-cells">
12          <div class="weui-cell">
13              <div class="weui-cell__bd"><!-- 输入框-->
14                  <input class="weui-input" type="text" placeholder="请输入文本"/>
15              </div>
16          </div>
17      </div>
18
19      <div class="weui-cells__title">性别</div>
20      <div class="weui-cells weui-cells__radio">
21          <label class="weui-cell weui-check__label" for="x11"><!-- 单选列表-->
22              <div class="weui-cell__bd">
23                  <p>男</p>
24              </div>
25              <div class="weui-cell__ft">
26                  <input type="radio" class="weui-check" name="radio1" id="x11">
27                  <span class="weui-icon-checked"></span>
28              </div>
29          </label>
```



```

30         <label class="weui-cell weui-check__label" for="x12">
31             <div class="weui-cell__bd">
32                 <p>女</p>
33             </div>
34             <div class="weui-cell__ft">
35                 <input type="radio" name="radio1" class="weui-check" id="x12"
checked="checked">
36                 <span class="weui-icon-checked"></span>
37             </div>
38         </label>
39     </div>
40     <div class="weui-cells__title">兴趣爱好</div>
41     <div class="weui-cells weui-cells_checkbox"><!-- 复选列表-->
42         <label class="weui-cell weui-check__label" for="s11">
43             <div class="weui-cell__hd">
44                 <input type="checkbox" class="weui-check" name="checkbox1" id="s11"
checked="checked">
45                 <i class="weui-icon-checked"></i>
46             </div>
47             <div class="weui-cell__bd">
48                 <p>旅游</p>
49             </div>
50         </label>
51         <label class="weui-cell weui-check__label" for="s12">
52             <div class="weui-cell__hd">
53                 <input type="checkbox" name="checkbox1" class="weui-check" id="s12">
54                 <i class="weui-icon-checked"></i>
55             </div>
56             <div class="weui-cell__bd">
57                 <p>音乐</p>
58             </div>
59         </label>
60         <label class="weui-cell weui-check__label" for="s13">
61             <div class="weui-cell__hd">

```

```

62         <input type="checkbox" name="checkbox1" class="weui-check" id="s13">
63         <i class="weui-icon-checked"></i>
64     </div>
65     <div class="weui-cell__bd">
66         <p>美术</p>
67     </div>
68 </label>
69 </div>
70 <div class="weui-cell weui-cell_vcode">
71     <div class="weui-cell__hd">
72         <label class="weui-label">手机号</label>
73     </div>
74     <div class="weui-cell__bd"> <!-- 手机号及验证-->
75         <input class="weui-input" type="tel" placeholder="请输入手机号">
76     </div>
77     <div class="weui-cell__ft">
78         <button class="weui-vcode-btn">获取验证码</button>
79     </div>
80 </div>
81 <div class="weui-cell weui-cell_select weui-cell_select-after">
82     <div class="weui-cell__hd">
83         <label for="" class="weui-label">国家/地区</label>
84     </div>
85     <div class="weui-cell__bd"> <!-- 选择列表项-->
86         <select class="weui-select" name="select2">
87             <option value="1">中国</option>
88             <option value="2">美国</option>
89             <option value="3">英国</option>
90         </select>
91     </div>
92 </div>
93 <label for="weuiAgree" class="weui-agree"> <!-- 复选框-->
94     <input id="weuiAgree" type="checkbox" class="weui-agree__checkbox">
95     <span class="weui-agree__text">

```



```

96             阅读并同意<a href="javascript:void(0);">《相关条款》</a>
97         </span>
98     </label>
99     <div class="weui-btn-area">
100     <a class="weui-btn weui-btn_primary" href="javascript:" id="showToolTips">确定</a>
101     </div>
102 </body>
103 </html>

```

【代码解释】

- 代码中使用了单选列表、复选列表、复选框等样式，代码已经注释清楚，这里不再详述。
- 要注意的是很多 Input 样式使用了双下画线，如第 94 行复选框样式是 `weui-agree__checkbox`。
- 除样式外，列表的使用跟 HTML 5 没有区别。

本例效果如图 10.8 所示。

The image shows a mobile web form with the following elements:

- 姓名 (Name):** A text input field with the placeholder text "请输入文本".
- 性别 (Gender):** A radio button group with options "男" (Male) and "女" (Female). The "女" option is selected, indicated by a checkmark.
- 兴趣爱好 (Hobbies):** A radio button group with options "旅游" (Travel), "音乐" (Music), and "美术" (Art). The "旅游" and "美术" options are selected, indicated by checkmarks.
- 手机号 (Mobile Number):** A text input field with the placeholder text "请输入手机号".
- 获取验证码 (Get Verification Code):** A button next to the mobile number field.
- 国家/地区 (Country/Region):** A dropdown menu currently showing "中国" (China).
- 阅读并同意《相关条款》 (Read and agree to the terms of service):** A checkbox that is checked.
- 确定 (Confirm):** A large button at the bottom of the form.

图 10.8 Input 效果

10.2.3 列表 (List)

微信提供的列表样式是 weui-cells，它还包括 5 种类型：

- 带说明的列表项。
- 带图标、说明的列表项。
- 带跳转的列表项。
- 带说明、跳转的列表项。
- 带图标、说明、跳转的列表项。

这里我们以带说明、跳转的列表项为例进行说明。新建 list.html，输入如下代码：

```

01  <!DOCTYPE html>
02  <html>
03  <head>
04      <meta charset="UTF-8">
05      <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=0">
06  <title 列表</title>
07  <link rel="stylesheet" href="style/weui.css"/>
08  </head>
09  <body>
10  <div class="weui-cells__title">将要跳转到</div>
11  <div class="weui-cells">
12      <a class="weui-cell weui-cell_access" href="javascript:window.location.href='signup.html';">
13          <div class="weui-cell__bd">
14              <p>注册</p>
15          </div>
16          <div class="weui-cell__ft">注册页面</div>
17      </a>
18      <a class="weui-cell weui-cell_access" href="javascript:window.location.href='welcome.html';">
19          <div class="weui-cell__bd">
20              <p>登录</p>
21          </div>
22          <div class="weui-cell__ft">登录页面</div>
23      </a>
24  </div>

```

```

25 </body>
26 </html>

```

【代码解释】

- 第 10 行给列表定义一个标题，使用了样式 `weui-cells__title`。
- 跳转使用 `href` 完成，在其中使用 `js` 语句 `window.location.href`。
- 列表项使用样式 `weui-cell__bd`，列表说明使用样式 `weui-cell__ft`。

本例效果如图 10.9 所示，点击注册后会跳转到注册页面。

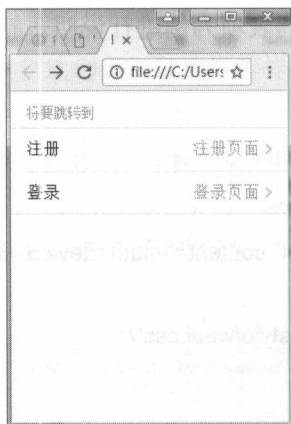


图 10.9 List 效果

10.2.4 滑块 (Slider)

微信提供的 Slider 有两种样式，一种只有滑动条，一种是带数字的滑动条。这里我们通过一个例子演示带数字的滑动条，并且滑动的同时，数字也会改变。

因为微信的操作大多都在手机端，所以使用样式库或者微信操作时，官方使用的库不是 jQuery，而是 zepto.js，所以本例也使用 zepto.js 实现数字的同时变化。新建 `slider.html` 文件，输入如下代码：

```

01 <!DOCTYPE html>
02 <html>
03 <head>
04     <meta charset="UTF-8">
05     <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=0">

```

```

06 <title 列表</title>
07 <link rel="stylesheet" href="style/weui.css"/>
08 <script src="example/zepto.min.js"></script>
09 <script type="text/javascript">
10     $(function(){
11         var $sliderTrack = $('#sliderTrack'),
12             $sliderHandler = $('#sliderHandler'),
13             $sliderValue = $('#sliderValue');
14
15         var totalLen = $('#sliderInner').width(),
16             startLeft = 0,
17             startX = 0;
18
19         $sliderHandler
20             .on('touchstart', function (e) {
21                 startLeft = parseInt($sliderHandler.css('left')) * totalLen / 100;
22                 startX = e.changedTouches[0].clientX;
23             })
24             .on('touchmove', function(e){
25                 var dist = startLeft + e.changedTouches[0].clientX - startX,
26                     percent;
27                 dist = dist < 0 ? 0 : dist > totalLen ? totalLen : dist;
28                 percent = parseInt(dist / totalLen * 100);
29                 $sliderTrack.css('width', percent + '%');
30                 $sliderHandler.css('left', percent + '%');
31                 $sliderValue.text(percent);
32                 e.preventDefault();
33             })
34         ;
35     });
36 </script>
37 </head>
38 <body>
39     <div class="page__hd">

```



```

40      <h1 class="page__title">Slider</h1>
41      <p class="page__desc">滑块</p>
42  </div>
43  <div class="page__bd page__bd_spacing">
44      <div class="weui-slider-box">
45          <div class="weui-slider">
46              <div id="sliderInner" class="weui-slider__inner">
47                  <div id="sliderTrack" style="width: 50%;" class="weui-slider__track">
48                      <div id="sliderHandler" style="left: 50%;" class="weui-slider__handler">
49                      </div>
50                  </div>
51                  <div id="sliderValue" class="weui-slider-box__value">50</div>
52              </div>
53          </div>
54      </body>
55  </html>

```

【代码解释】

- 在第 8 行引入 zepto.min.js 库。
- 第 20~33 行为滑块添加 touchstart 和 touchmove 事件，以保证滑块滑动的同时数字也发生变化。
- 第 47、48 行设置滑块默认显示在中间。注意，这里用 50% 宽度而不是固定宽度是为了保证各移动端的兼容，宽度不会溢出。

本例效果如图 10.10 所示，要注意操作效果必须在移动端测试，也可以在 Chrome 浏览器中模拟手机端。

10.2.5 上传 (Uploader)

Uploader 在微信中用于上传图片，一般与 Gallery 组合使用，Gallery 用于删除上传的图片。

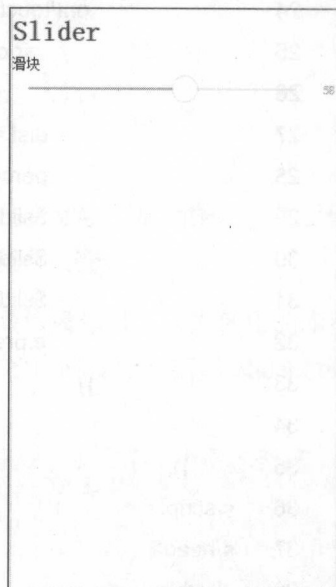


图 10.10 Slider 效果

我们来设计一个允许上传图片 and 删除图片的例子。新建 `uploader.html`，输入代码如下，这里也使用了 `zepto.js` 库来为组件添加事件。

```
01  <!DOCTYPE html>
02  <html>
03  <head>
04      <meta charset="UTF-8">
05      <meta name="viewport" content="width=device-width,initial-scale=1,user-scalable=0">
06  <title>上传</title>
07  <link rel="stylesheet" href="style/weui.css"/>
08  <script src="example/zepto.min.js"></script>
09  <script type="text/javascript">
10      $(function(){
11          var tpl = '<li class="weui-uploader__file" style="background-image:url(#url#)"></li>',
12              $gallery = $("#gallery"), $galleryImg = $("#galleryImg"),
13              $uploaderInput = $("#uploaderInput"),
14              $uploaderFiles = $("#uploaderFiles")
15          ;
16          $uploaderInput.on("change", function(e){
17              var src, url = window.URL || window.webkitURL || window.mozURL, files = e.target.files;
18              for (var i = 0, len = files.length; i < len; ++i) {
19                  var file = files[i];
20                  if (url) {
21                      src = url.createObjectURL(file);
22                  } else {
23                      src = e.target.result;
24                  }
25                  $uploaderFiles.append($(tpl.replace('#url#', src)));
26              }
27          });
28          $uploaderFiles.on("click", "li", function(){
29              $galleryImg.attr("style", this.getAttribute("style"));
30              $gallery.fadeIn(100);
31          });
32          $gallery.on("click", function(){
```



```
33     $gallery.fadeOut(100);
34   });
35 });
36 </script>
37 </head>
38 <body>
39   <div class="weui-gallery" id="gallery">
40     <span class="weui-gallery__img" id="galleryImg"></span>
41     <div class="weui-gallery__opr">
42       <a href="javascript:" class="weui-gallery__del">
43         <i class="weui-icon-delete weui-icon_gallery-delete"></i>
44       </a>
45     </div>
46   </div>
47   <div class="weui-uploader">
48     <div class="weui-uploader__hd">
49       <p class="weui-uploader__title">图片上传</p>
50       <div class="weui-uploader__info">0/2</div>
51     </div>
52     <div class="weui-uploader__bd">
53       <ul class="weui-uploader__files" id="uploaderFiles">
54         <li class="weui-uploader__file" style="background-image:url
(example/images/pic_160.png)"></li>
55         <li class="weui-uploader__file" style="background-image:url
(example/images/pic_160.png)"></li>
56         <div class="weui-uploader__file-content">
57           <i class="weui-icon-warn"></i>
58         </div>
59       </li>
60       <li class="weui-uploader__file weui-uploader__file_status" style=
"background-image:url(example/images/pic_160.png)">
61         <div class="weui-uploader__file-content">50%</div>
62       </li>
63     </ul>
```

```

64         <div class="weui-uploader__input-box">
65             <input id="uploaderInput" class="weui-uploader__input" type="file" accept=
"image/*" multiple/>
66         </div>
67     </div>
68 </div>
69 </body>
70 </html>

```

【代码解释】

- 第 53~63 行设计一个图片列表，使用了样式 `weui-uploader__files`，这是微信中专用的图片列表形式。
- 第 65 行设计上传按钮，应用了样式 `weui-uploader__input`。
- 第 28~34 行分别为图片列表和删除组件添加 `click` 事件。
- 第 39~45 行定义了一个回收框，默认情况下在界面中看不出来，只有在调用时才显示。

本例效果如图 10.11 所示，读者可以看到，在微信中上传图片有固定的格式，都是一个方框一个方框的，这和我们普通网页中的上传非常不一样。当我们单击上传后的图片，会显示如图 10.12 的效果，这是 Gallery 的作用，单击下面的回收框图标，就会删除上传后的图片，这里只展示了一个动画效果。

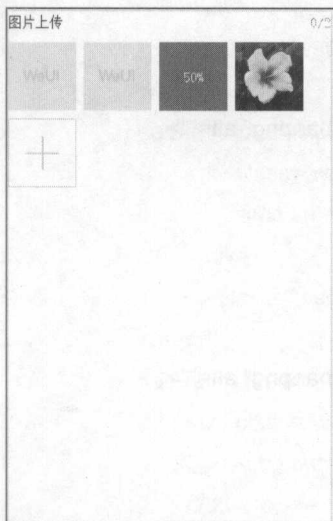


图 10.11 图片上传效果

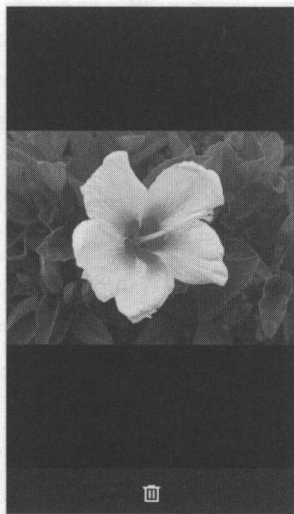


图 10.12 图片删除效果

10.3 使用基础组件

基础组件包括了章标题、进度条、页脚、表格、回收框等。在上一节最后我们上传组件中使用了回收框，读者可以看到其使用方式跟大多数表单组件也是一样的。既然一样，我们就不逐个介绍了，本节介绍几个常用的组件。

10.3.1 九宫格（Grid）

微信调用网页的时候，一般都不使用网页常见的导航栏，而是使用九宫格来显示网页提供的各种功能，这个我们在各种交通公众号、银行公众号中常常见到。本例就使用微信的 Grid 样式实现一个九宫格效果。

新建 grid.html，输入代码如下：

```
01 //省略部分框架代码
02 <div class="weui-grids">
03     <a href="javascript:;" class="weui-grid">
04         <div class="weui-grid__icon">
05             
06         </div>
07         <p class="weui-grid__label">新闻</p>
08     </a>
09     <a href="javascript:;" class="weui-grid">
10         <div class="weui-grid__icon">
11             
12         </div>
13         <p class="weui-grid__label">体育</p>
14     </a>
15     <a href="javascript:;" class="weui-grid">
16         <div class="weui-grid__icon">
17             
18         </div>
19         <p class="weui-grid__label">娱乐</p>
20     </a>
21     <a href="javascript:;" class="weui-grid">
```

```

22         <div class="weui-grid__icon">
23             
24         </div>
25         <p class="weui-grid__label">视频</p>
26     </a>
27     <a href="javascript:;" class="weui-grid">
28         <div class="weui-grid__icon">
29             
30         </div>
31         <p class="weui-grid__label">网红</p>
32     </a>
33     <a href="javascript:;" class="weui-grid">
34         <div class="weui-grid__icon">
35             
36         </div>
37         <p class="weui-grid__label">社会</p>
38     </a>
39     <a href="javascript:;" class="weui-grid">
40         <div class="weui-grid__icon">
41             
42         </div>
43         <p class="weui-grid__label">生活</p>
44     </a>
45     <a href="javascript:;" class="weui-grid">
46         <div class="weui-grid__icon">
47             
48         </div>
49         <p class="weui-grid__label">地方</p>
50     </a>
51     <a href="javascript:;" class="weui-grid">
52         <div class="weui-grid__icon">
53             
54         </div>
55         <p class="weui-grid__label">身边</p>

```



```
56         </a>
```

```
57     </div>
```

【代码解释】

- 九宫格的每一格的组成都是固定的<a>、<div>、和<p>，样式也是由 weui-grid、weui-grid__icon 和 weui-grid__label 组成的。
- 每一格的跳转通过 href 完成。

本例效果如图 10.13 所示。

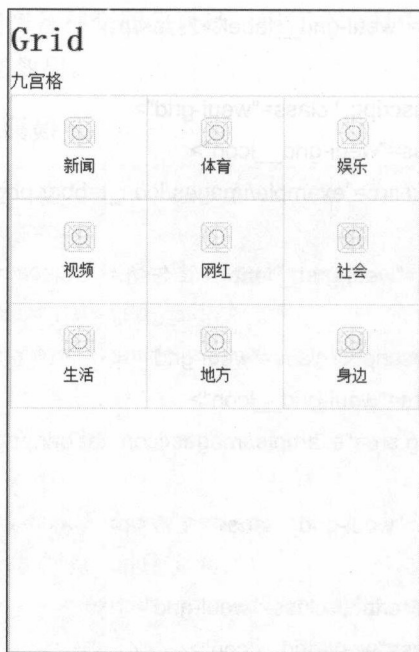


图 10.13 九宫格效果

10.3.2 进度条 (Progress)

在上传图片的时候一般会出现进度条，而且在上传多个图片的时候会出现多个进度条，微信的进度条效果也是这样的，如图 10.14 所示。

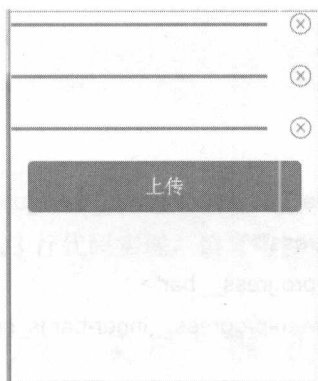


图 10.14 进度条效果

进度条的使用也要配合 zepto.js 库来完成。新建 progress.html 文件，输入代码如下：

```

01  <script type="text/javascript">
02      $(function(){
03          var $progress = $('#js_progress'),
04              $btnUpload = $('#btnUpload');
05          var progress = 0;
06
07          function next() {
08              if(progress > 100){
09                  progress = 0;
10                  $btnUpload.removeClass('weui-btn_disabled');
11                  return;
12              }
13              $progress.css({width: progress + '%'});
14              progress = ++progress;
15              setTimeout(next, 20);
16          }
17
18          $btnUpload.on('click', function(){
19              if ($btnUpload.hasClass('weui-btn_disabled')) return;
20
21              $btnUpload.addClass('weui-btn_disabled');
22              next();

```



```
23         });
24     });
25 </script>
26 <body>
27 <div class="page__bd page__bd_spacing">
28     <div class="weui-progress">
29         <div class="weui-progress__bar">
30             <div class="weui-progress__inner-bar js_progress" style="width: 0%;"></div>
31         </div>
32         <a href="javascript:;" class="weui-progress__opr">
33             <i class="weui-icon-cancel"></i>
34         </a>
35     </div>
36     <br>
37     <div class="weui-progress">
38         <div class="weui-progress__bar">
39             <div class="weui-progress__inner-bar js_progress" style="width: 50%;"></div>
40         </div>
41         <a href="javascript:;" class="weui-progress__opr">
42             <i class="weui-icon-cancel"></i>
43         </a>
44     </div>
45     <br>
46     <div class="weui-progress">
47         <div class="weui-progress__bar">
48             <div class="weui-progress__inner-bar js_progress" style="width: 80%;"></div>
49         </div>
50         <a href="javascript:;" class="weui-progress__opr">
51             <i class="weui-icon-cancel"></i>
52         </a>
53     </div>
54     <div class="weui-btn-area">
55         <a href="javascript:;" class="weui-btn weui-btn_primary" id="btnUpload">上传</a>
56     </div>
```

```
57 </div>
58 </body>
```

【代码解释】

- 进度条的样式是 weui-progress_bar。
- 上传的事件通过第 18~23 行代码实现，读者可以更改这里看看效果。

10.4 使用导航组件

微信提供的导航组件有两个：Navbar（效果图如图 10.15 所示）和 Tabbar（效果图如图 10.16 所示）。

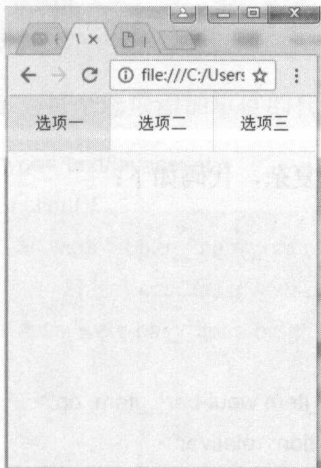


图 10.15 Navbar 效果

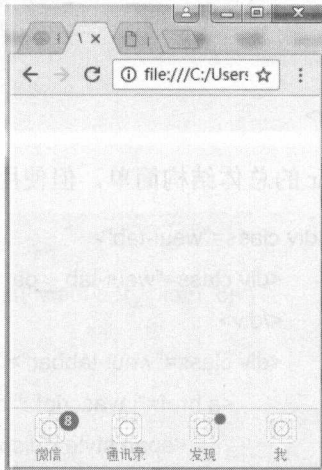


图 10.16 Tabbar 效果

Navbar 比较简单，使用几个 div 即可实现，代码如下：

```
01 <div class="weui-tab">
02   <div class="weui-navbar">
03     <div class="weui-navbar__item weui-bar__item_on">
04       选项一
05     </div>
06     <div class="weui-navbar__item">
07       选项二
08     </div>
```

```

09         <div class="weui-navbar__item">
10             选项三
11         </div>
12     </div>
13 </div>

```

如果需要为每个选项添加 click 事件，只需增加如下代码：

```

<script type="text/javascript">
    $(function(){
        $(''.weui-navbar__item').on('click', function () {
            $(this).addClass('weui-bar__item_on').siblings(''.weui-bar__item_on').
removeClass('weui-bar__item_on');
        });
    });
</script>

```

Tabbar 的总体结构简单，但使用 span 的样式稍显复杂，代码如下：

```

01 <div class="weui-tab">
02     <div class="weui-tab__panel">
03     </div>
04     <div class="weui-tabbar">
05         <a href="javascript:;" class="weui-tabbar__item weui-bar__item_on">
06             <span style="display: inline-block;position: relative;">
07                 
08                 <span class="weui-badge" style="position: absolute;top: -2px;right:
-13px;">8</span>
08             </span>
09             <p class="weui-tabbar__label">微信</p>
10         </a>
11         <a href="javascript:;" class="weui-tabbar__item">
12             
13             <p class="weui-tabbar__label">通讯录</p>
14         </a>
15         <a href="javascript:;" class="weui-tabbar__item">

```

```

16      <span style="display: inline-block;position: relative;">
17          
18          <span class="weui-badge weui-badge_dot" style="position: absolute;top:
0;right: -6px;"></span>
19      </span>
20      <p class="weui-tabbar__label">发现</p>
21  </a>
22  <a href="javascript:;" class="weui-tabbar__item">
23      
24      <p class="weui-tabbar__label">我</p>
25  </a>
26  </div>
27 </div>

```

Tabbar 为每个选项添加事件的代码如下:

```

<script type="text/javascript">
    $(function(){
        $('.weui-tabbar__item').on('click', function () {
            $(this).addClass('weui-bar__item_on').siblings('.weui-bar__item_on')
.removeClass('weui-bar__item_on');
        });
    });
</script>

```

10.5 网页其他样式

除了前面介绍的样式以外, 微信还提供了操作反馈和搜索相关的样式。操作反馈有弹出式菜单 Actionsheet、弹出式提示 Toast、对话框 Dialog、提示页 Msg 和多列选择器 Picker。搜索相关的样式只有一个 Search Bar, 其实就是调出搜索栏。这些样式的使用和前面介绍的大体相同, 所以就不一一介绍了。

10.6 常见问题

1. 没学过 zepto.js 怎么办？

微信官方提供的网页开发样式全都使用的是 zepto.js 库，但这并不是必需的，如果我们有其他习惯使用的库，如 jQuery 或 jQuery Mobile，也是可以的，主要的操作是为表单中的组件添加事件，基本常用的库都是可以的。

2. 这是网页样式，和微信开发有啥关系？

虽然这个库的名字是网页开发样式库，但有个前提是“微信网页”，即应用在微信中的网页上的样式。本例中这些网页，都可以直接上传到服务器，然后用在“阅读原文”或指定自定义菜单的 URL 上。

第 11 章 实战案例 1：中国好友挑战赛（HTML 5 混编）



从本章开始，将讲解微信 JSSDK 在实战中的运用。实际生活中的好多项目是不区分微信体系内，还是微信体系外的 Web 应用。那么，不在微信体系内的 Web 应用就必须寻找其他的替代性方案了。

微信在商业活动中最常见的需求就是分享到朋友圈、分享给 QQ 好友这类传播式需求，借用互联网的力量，迅速将公司产品的口碑、价值以及对用户的作用，告诉用户。

11.1 案例概述

本项目来自一个真实的甲方需求，内容有所改动。客户想实现一个基于 HTML 5、JavaScript 技术的微信宣传类型的项目。游戏的业务是，用户首先打开首页，然后选择自己是男生还是女生，点击进入对应性别的游戏主页，然后玩游戏。其中女生页是按住小圆圈，男生页是握住手机底部，滑动拇指，最后分别显示女生或男生的游戏结果，用户可以将结果分享到朋友圈、分享给 QQ 好友等。如果用户对这个宣传小游戏感兴趣，再引导用户玩高级模式，即引导用户下载 APP。

11.2 客户需求

11.2.1 首页

先看看首页的介绍，如图 11.1~图 11.3 所示。

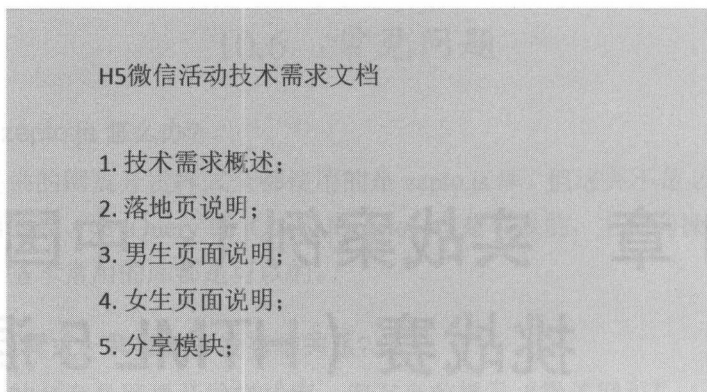


图 11.1 整体说明

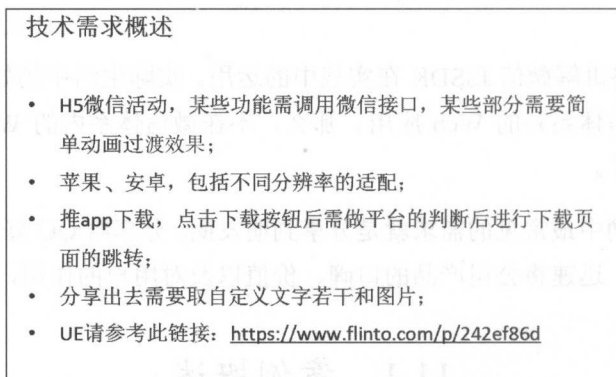


图 11.2 技术需求概述

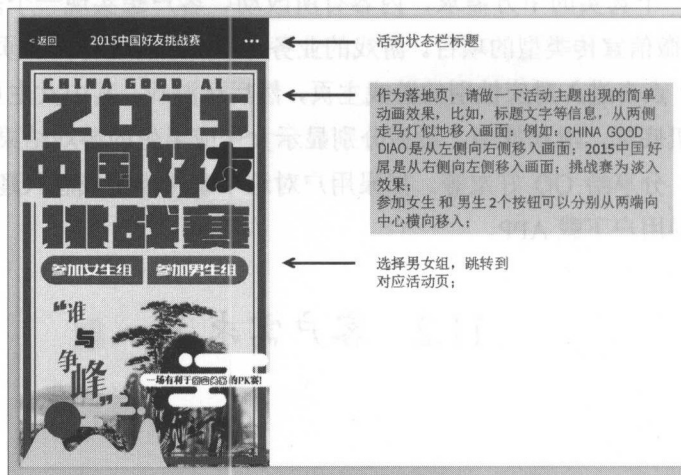


图 11.3 落地页(首页)

11.2.2 男生游戏的逻辑

再看看男生游戏的逻辑页面, 如图 11.4~图 11.10 所示。

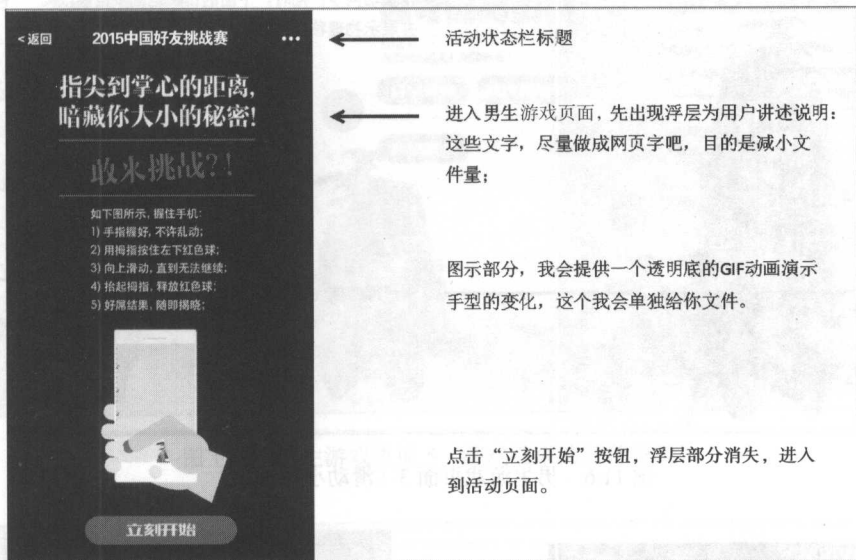


图 11.4 男生游戏页面 1 (开始页)

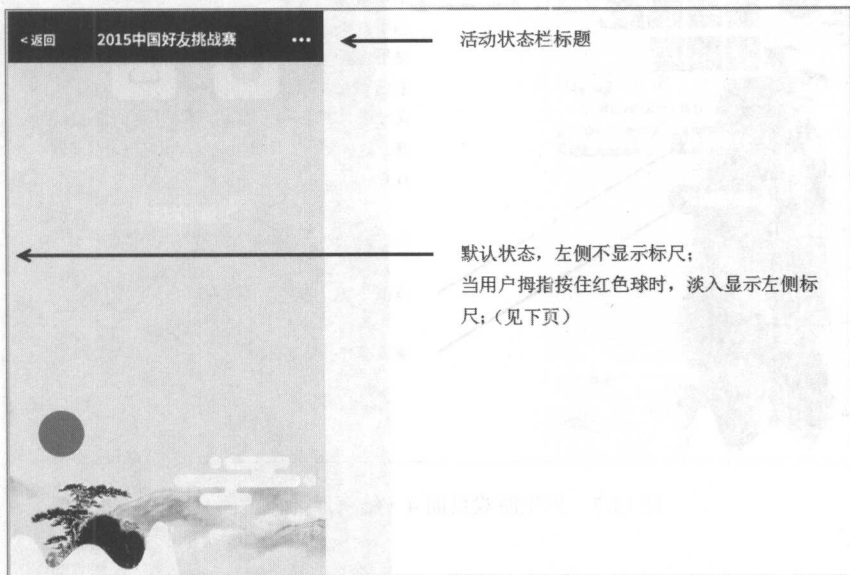


图 11.5 男生游戏页面 2 (初始页面)

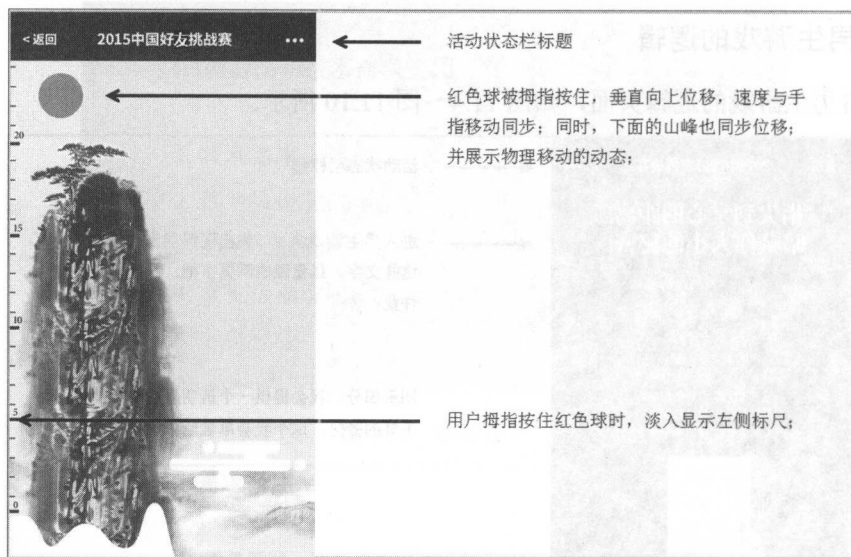


图 11.6 男生游戏页面 3（滑动小球结束）

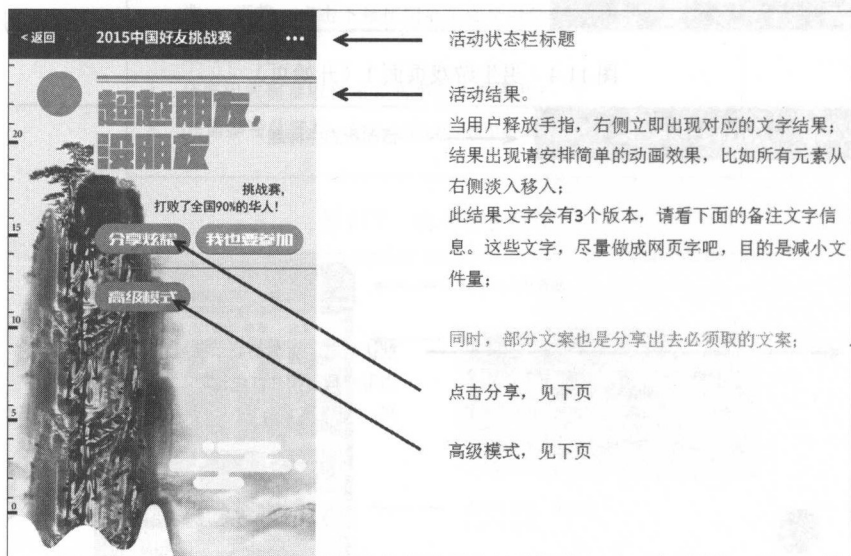


图 11.7 男生游戏页面 4（结束后弹框页面）



图 11.8 男生游戏页面 5（游戏结束的三种表现）

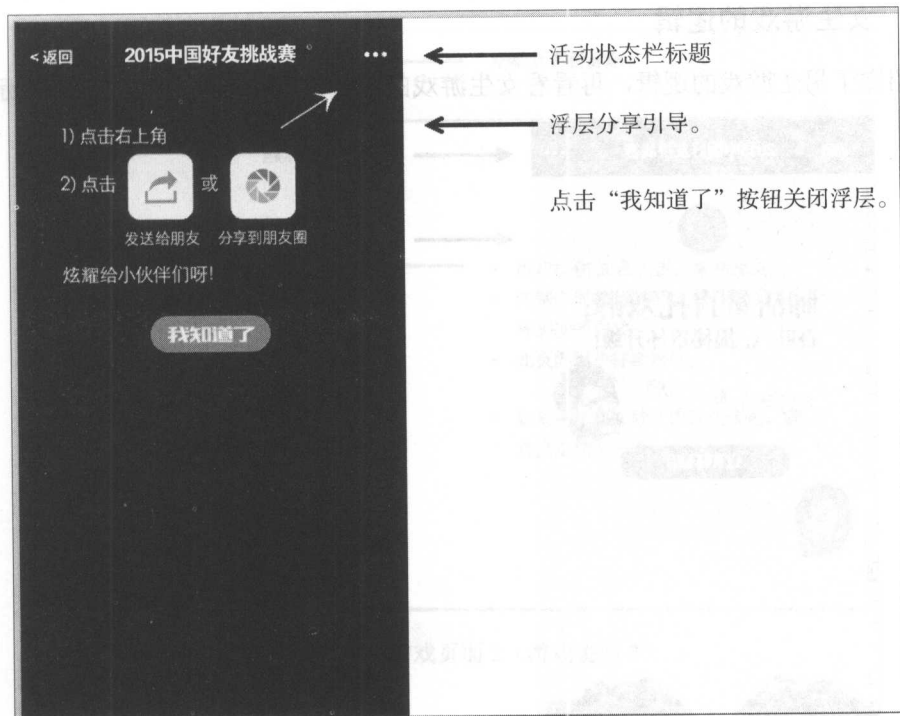


图 11.9 男生游戏页面 6（用户分享页面）

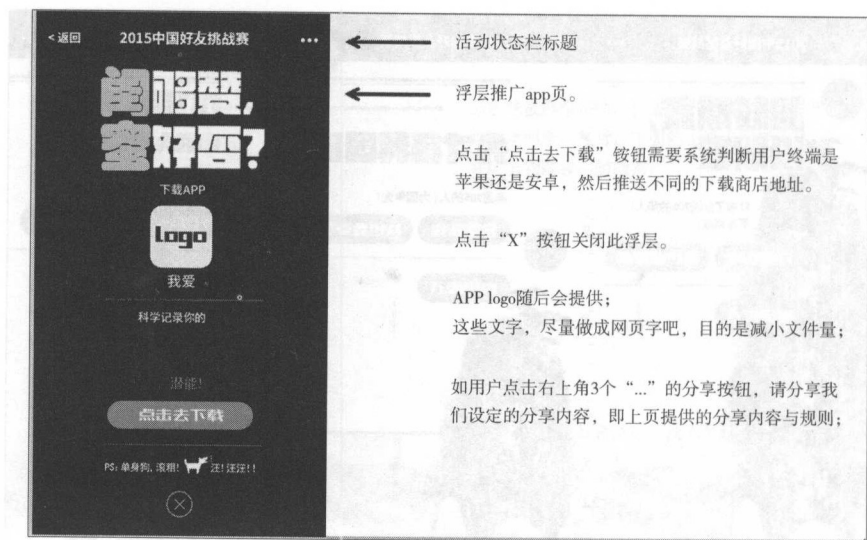


图 11.10 男生游戏页面 7（用户下载 APP）

11.2.3 女生游戏的逻辑

介绍完了男生游戏的逻辑，再看看女生游戏的逻辑，如图 11.11～图 11.20 所示。

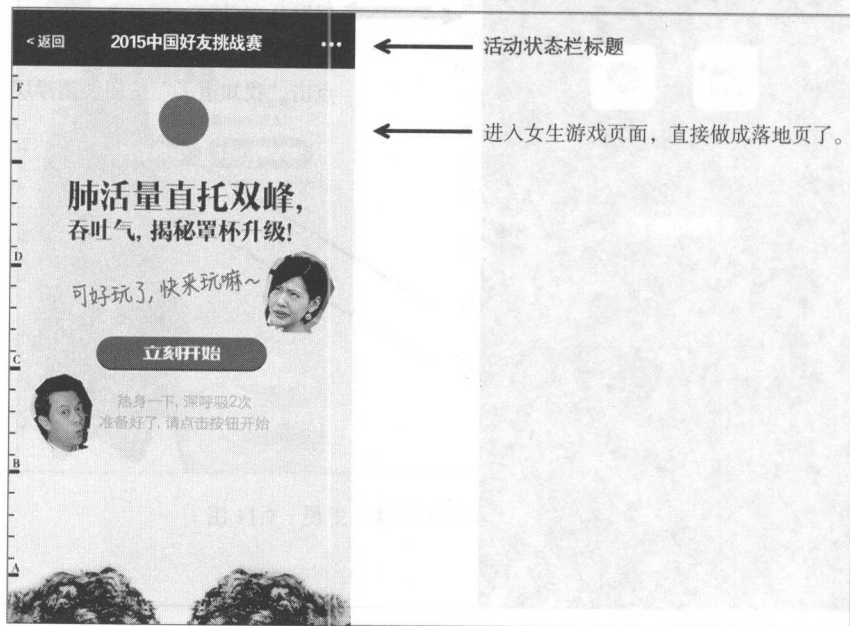


图 11.11 女生游戏页面 1（开始游戏）

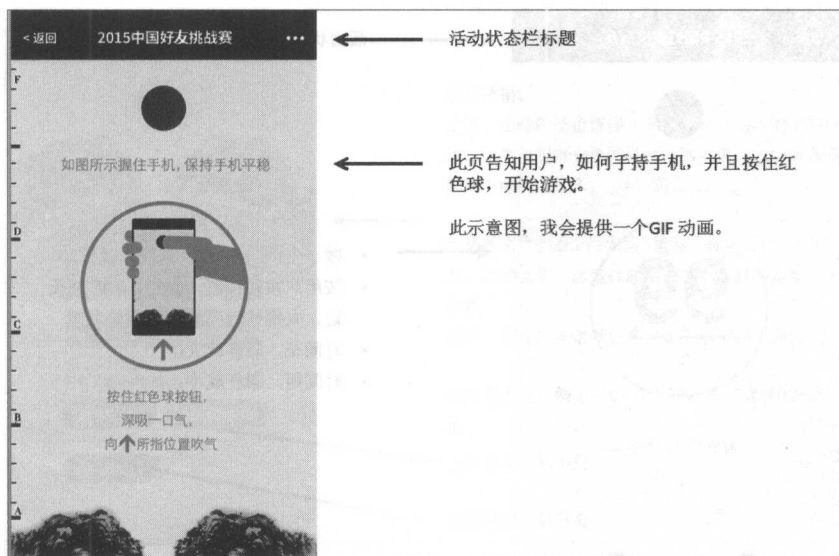


图 11.12 女生游戏页面 2 (游戏说明 1)

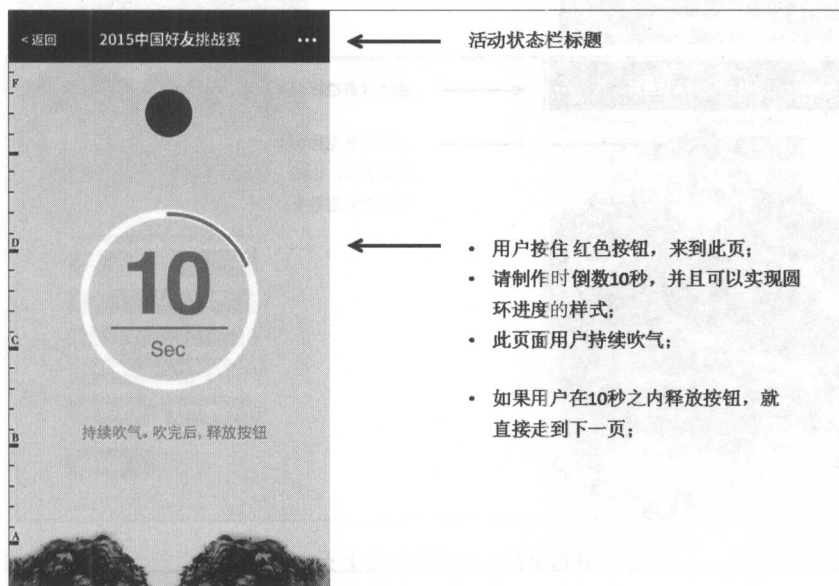


图 11.13 女生游戏页面 2 (游戏说明 2)

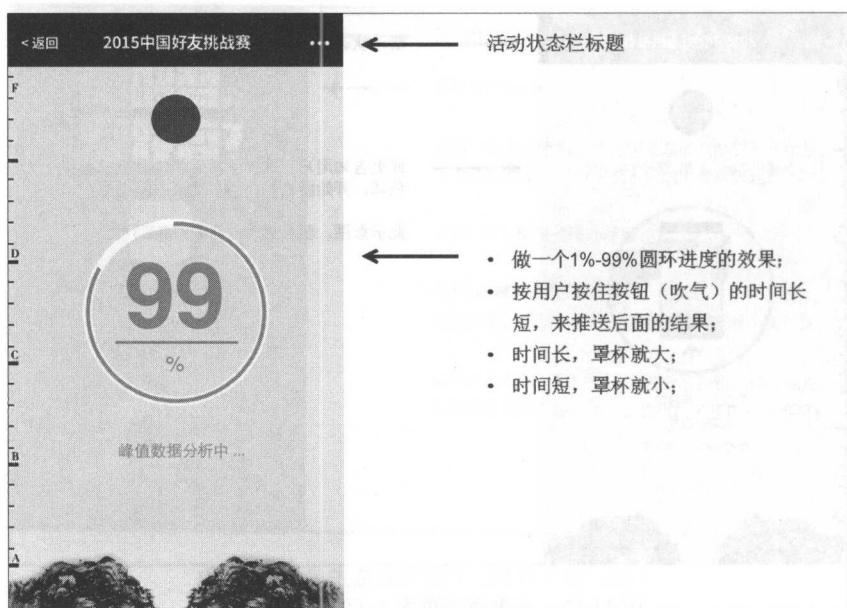


图 11.14 女生游戏页面 3（游戏说明 3）

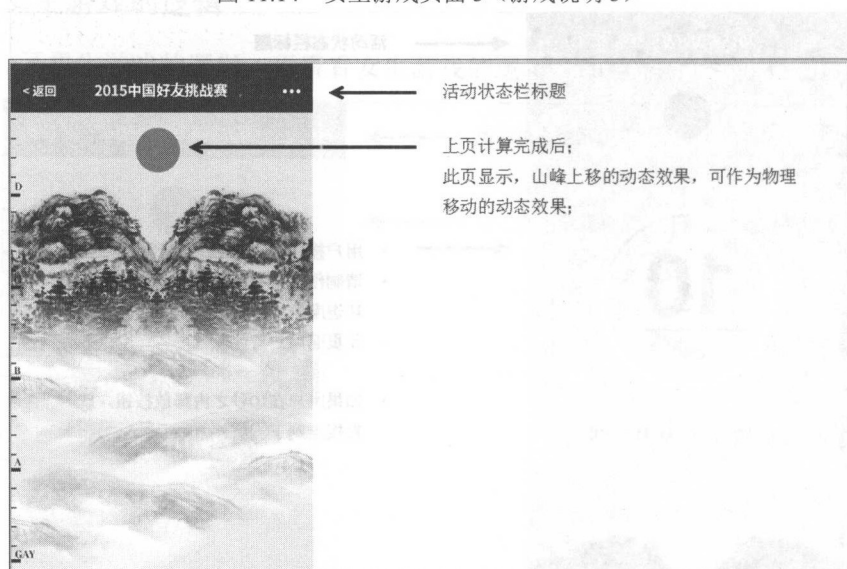


图 11.15 女生游戏页面 4（游戏说明 4）

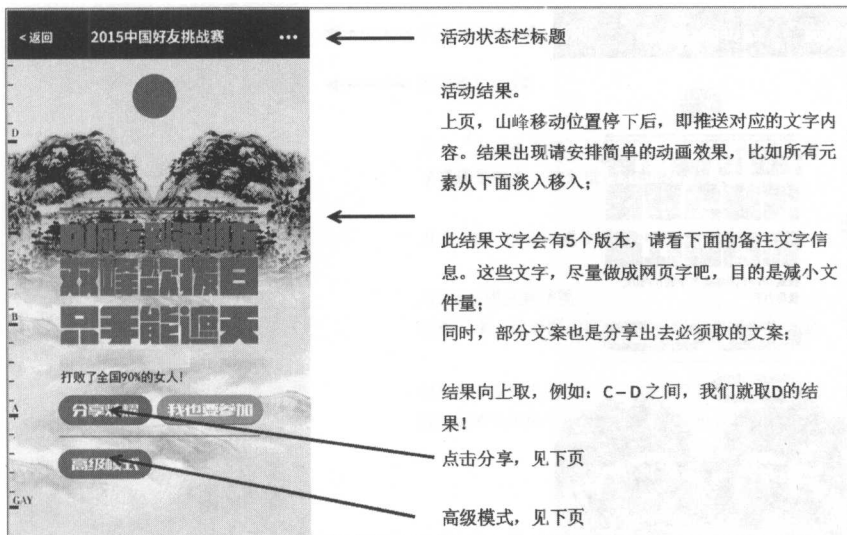


图 11.16 女生游戏页面 5 (结果页说明 1)



图 11.17 女生游戏页面 6 (结果页说明 2)



图 11.18 女生游戏页面 7 (结果页说明 3)

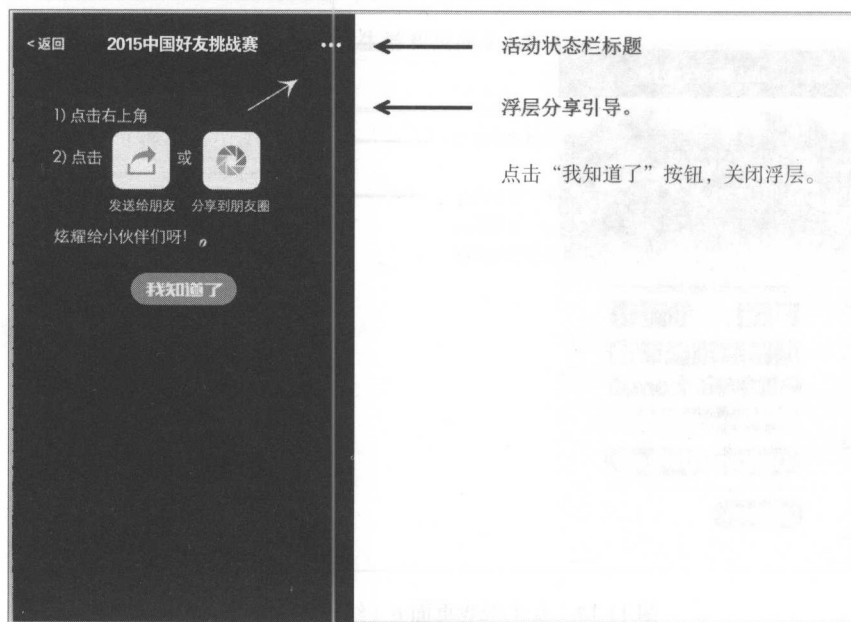


图 11.19 女生游戏页面 8 (分享页)

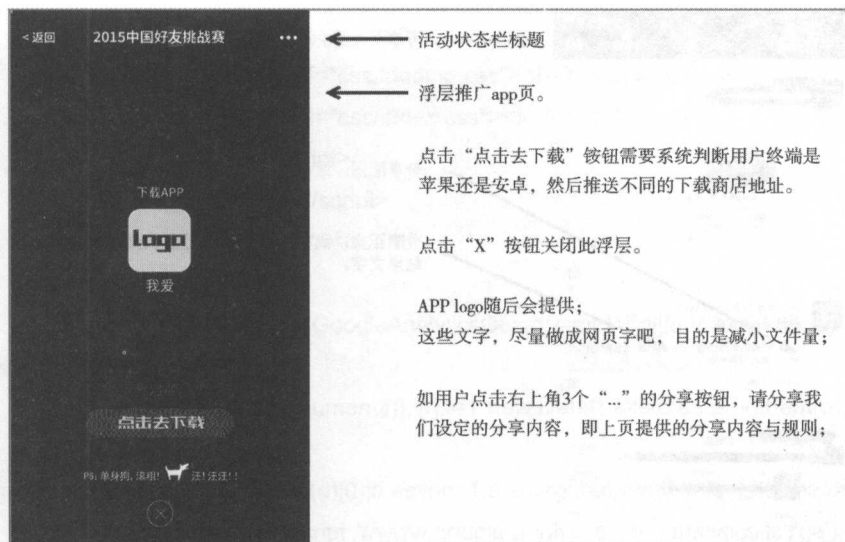


图 11.20 女生游戏页面 9 (APP 下载页)

11.2.4 分享页面

最后，看看分享页面，如图 11.21 和图 11.22 所示。

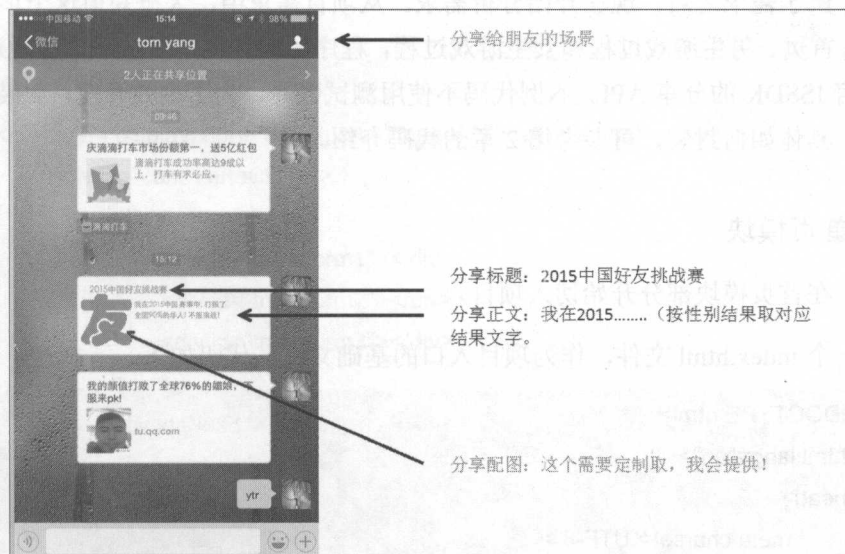


图 11.21 分享页演示 1

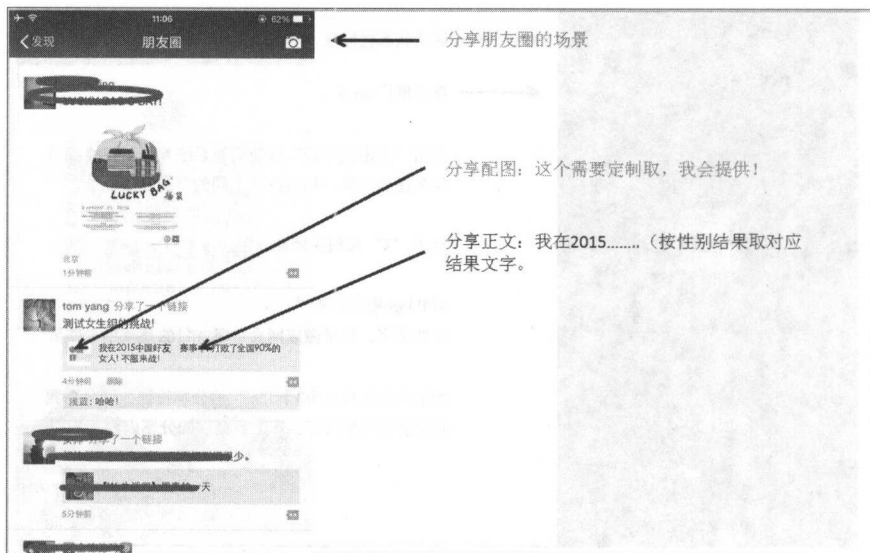


图 11.22 分享页演示 2

11.3 实现代码及分析

详细了解了需求之后, 现在开始分析需求。从项目需求中, 大概知道这个项目分 3 个模块, 包含首页、男生游戏过程和女生游戏过程, 程序的构建也可以按照这样的逻辑划分。涉及到微信 JSSDK 的分享 API。本例代码不使用测试账号, 为了高效开发, 封装了微信提供的接口, 具体如何封装, 可参考第 2 章的代码介绍。

11.3.1 首页模块

第一, 在首页模块部分开始切入项目。

创建一个 index.html 文件, 作为项目入口的基础文件, 代码如下:

```
01 <!DOCTYPE html>
02 <html lang="en">
03 <head>
04     <meta charset="UTF-8">
05     <title>我在 2015 “中国好友” 挑战赛, 不服来撕 (战) </title>
06     <meta name="description" content="我在 2015 “中国好友” 挑战赛, 不服来撕 (战) "/>
```

```

07     <meta name="viewport" content="initial-scale=1, user-scalable=no">
08     <link rel="stylesheet" href="css/loading.css"><!--预加载样式表-->
09     <link rel="stylesheet" href="css/index.css"><!--入口页样式表-->
10     <script src="js/jq.js"></script>
11     <script src="js/index.js"></script>
12     <script src="js/viewport.js"></script>
13     <script>
14         (function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){//Google 监控代码
15 检测用户的一些访问数据
16             (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o),
17
18             m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)
19             })(window,document,'script','/www.google-analytics.com/analytics.js','ga');
20
21             ga('create', 'UA-60790343-1', 'auto');
22             ga('send', 'pageview');
23
24     </script>
25
26 </head>
27 <body>
28 <div style="display: none"></div>
29     <div id="caseVerteClaire">
30         <div id="transform">
31             <div id="transform1"></div>
32             <div id="transform2"></div>
33             <div id="transform3"></div>
34         </div>
35         <div id="load">
36             <p>loading...</p>
37         </div>
38     </div>
39     <div class="wrap">
40         

```

```
41      <!-- 浮动层-->
42      <div class="wrap_con">
43          
45          
47          
49          
51          <div class="btns">
52              
53              
55          </div>
56
57      </div>
58
59      
60      <div class="index_bg1">
61          
62          
63          
64      </div>
65  </div>
66  <script>
67  var _hmt = _hmt || [];
68  (function() {
69      var hm = document.createElement("script");
70      hm.src = "//hm.baidu.com/hm.js?c740d6849c73e736e8ad06064c731909";
71      var s = document.getElementsByTagName("script")[0];
72      s.parentNode.insertBefore(hm, s);
73  })();
74  </script>
```



```
75 </body>
76 </html>
```

在 index.html 文件中, 引入了几个依赖文件及库, 用于快速开发。它包含两个样式表, “loading.css” 是预加载的 UI 布局, “index.css” 是当前页的 UI 布局; 以及 3 个 JS 文件: “jq.js” 是辅助简化 js 开发的库, “index.js” 是当前模块的主要业务逻辑, “viewport.js” 缩放视口适配用, 其中 “jq.js”、“viewport.js” 这两个库, 可以到官网下载或在示例源代码中查看。

新建 “loading.css” 文件, 代码如下:

```
01  /** loading***/
02  #caseVerteClaire {
03      height : 100%;
04      width : 100%;
05      position : absolute;
06      background: rgba(0,0,0,0.7);
07      z-index: 10;
08  }
09  #caseVerteClaire #load {
10      color : #fff;
11      font-family : calibri;
12      text-align : center;
13      position : absolute;
14      font-size: 40px;
15      top: 31%;
16      left: 26%;
17  }
18
19  #caseVerte #case5 {
20      height : 10px;
21      width : 10px;
22      background-color : #fbfbfb;
23      -webkit-animation : case3 1.5s infinite;
24  }
```



```
25
26 #caseBleu #case5 {
27     height : 10px;
28     width : 10px;
29     background-color : #0086a6;
30     -webkit-animation : case3 1.5s infinite;
31 }
32
33 @-webkit-keyframes case5 {
34     0% {width : 0%;}
35     50% {width : 105px;}
36     100% {width : 0%;}
37 }
38 #transform {
39     position: absolute;
40     top: 30%;
41     left: 30%;
42 }
43
44 #transform1 {
45     height : 30px;
46     width : 30px;
47     border-radius : 50%;
48     background-color : #fff;
49     position : absolute;
50     top : 85px;
51     left : 15px;
52     opacity : 0;
53     animation : transform 4s infinite;
54     -webkit-animation : transform 4s infinite;
55 }
56
57 #transform2 {
58     height : 30px;
```

```
59     width : 30px;
60     border-radius : 50%;
61     background-color : #fff;
62     position : absolute;
63     top : 85px;
64     left : 54px;
65     opacity : 0;
66     animation : transform 4s infinite;
67     -webkit-animation : transform 4s infinite;
68     animation-delay : 0.5s;
69     -webkit-animation-delay : 0.5s;
70 }
71
72 #transform3 {
73     height : 30px;
74     width : 30px;
75     border-radius : 50%;
76     background-color : #fff;
77     position : absolute;
78     top : 85px;
79     left : 94px;
80     opacity : 0;
81     animation : transform 4s infinite;
82     -webkit-animation : transform 4s infinite;
83     animation-delay : 1s;
84     -webkit-animation-delay : 1s;
85 }
86
87 @-webkit-keyframes transform {
88     0% {opacity : 0;}
89     25% {opacity : 1;}
90     50% {opacity : 0;}
91     75% {opacity : 1;}
192    100% {opacity : 0;}
```

```
193 }
194
195 @keyframes transform {
196     0% {border-radius : 0px; opacity : 0;}
197     20% {border-radius : 0px; opacity : 1;}
198     40% {border-radius : 0px; opacity : 0;}
199     60% {border-radius : 50%; opacity : 0;}
200     80% {border-radius : 50%; opacity : 1;}
201     100% {border-radius : 50%; opacity : 0;}
202 }
203 .none{
204     display: none;
205 }
206 /*** 吹起圆形样式 ***/
207 .pie {
208     width:200px;
209     height:200px;
210     background-color:#fff;
211     border-radius:100px;
212     position:absolute;
213 }
214 .pie1 {
215     clip:rect(0px,200px,200px,100px);
216     /*-o-transform:rotate(0deg);
217     -moz-transform:rotate(0deg);
218     -webkit-transform:rotate(0deg);
219     background:-moz-radial-gradient(20% 50% 0deg, #333, #0000ff);
220     background:-webkit-gradient(radial, 100 100, 0, 100 100, 110, from(#000), to(#0000ff)); */
221 }
222 .pie2 {
223     clip:rect(0px,100px,200px,0px);
224     /*-o-transform:rotate(0deg);
225     -moz-transform:rotate(0deg);
226     -webkit-transform:rotate(0deg);
```



```
127     background:-moz-radial-gradient(80% 50% 0deg, #333, #0000ff);
128     background:-webkit-gradient(radial, 100 100, 0, 100 100, 110, from(#000), to(#0000ff)); */
129 }
130 .hold {
131     width:200px;
132     height:200px;
133     position:absolute;
134     z-index:1;
135 }
136 .hold1 {
137     clip:rect(0px,200px,200px,100px);
138 }
139 .hold2 {
140     clip:rect(0px,100px,200px,0px);
141 }
142 .bg {
143     width:200px;
144     height:200px;
145     background-color:#3ab7ff;
146     border-radius:100px;
147     position:absolute;
148     box-shadow:0px 0px 8px #333;
149     /* background:-moz-radial-gradient(0% 50% 0deg, #900, #ff0000);
150     background:-webkit-gradient(radial, 100 100, 0, 100 100, 110, from(#900), to(#ff0000)); */
151 }
152 .time {
153     width:170px;
154     height:170px;
155     margin:15px 0 0 15px;
156     background-color:#fff;
157     border-radius:100px;
158     position:absolute; z-index:1;
159     box-shadow:0px 0px 8px #333 inset;
160     text-align:center;
```

```
161     line-height:160px;
162     font-size:35px;
163     font-weight:bold;
164 }
165 .time .em{
166     float: left;
167     width:140px;
168     opacity:1;
169     color: #3ab7ff;
170     font-size: 1.1em;
171     margin-left: 12px;
172     margin-top: 10px;
173 }
174 .em strong,.em em{
175     float: left;
176     width: 100%;
177     height: 70px;
178     line-height: 70px;
179 }
180 .em em{
181     font-weight: normal;
182     font-style: normal;
183 }
184 .em strong{
185     text-shadow: 1px 1px 3px #333;
186     border-bottom: 2px solid #3ab7ff;
187 }
```

新建“index.css”文件，代码如下：

```
01 @charset 'utf-8';
02 /* CSS reset */
03 html{color:#000;background:#FFF;font-family:'Microsoft YaHei',Arial, Helvetica, sans-serif;}
04 body,div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,h5,h6,pre,code,form,fieldset,legend,a,input,button,textarea
05 a,p,blockquote,th,td,strong{padding:0;margin:0;font-family:'Microsoft YaHei',Arial, Helvetica,
```



```
06  sans-serif;}
07  table{border-collapse:collapse;border-spacing:0;}
08  fieldset,img{border:0;}
09  a{}/*此处待添加默认链接颜色*/
10  var,em,strong{font-style:normal;}
11  address,caption,cite,code,dfn,em,strong,th,var, optgroup{font-style:inherit;font-weight:inherit;}
12  del,ins{text-decoration:none;}
13  li{list-style:none;}
14  caption,th{text-align:left;}
15  h1,h2,h3,h4,h5,h6{font-size:100%;font-weight:normal;}
16  q:before,q:after{content:"";}
17  abbr,acronym{border:0;font-variant:normal;}
18  sup{vertical-align:baseline;}
19  sub{vertical-align:baseline;}
20  legend{color:#000;}
21  input,button,textarea,select,optgroup,option{font-family:inherit;
22  font-size:inherit;font-style:inherit;font-weight:inherit;}
23  input,button,textarea,select{*font-size:100%;}
24  body{-webkit-user-select:none;-webkit-text-size-adjust:none;
25  -webkit-transform-style:preserve-3d;}
26  *{-webkit-tap-highlight-color:rgba(0,0,0,0);}
27  /*** clear style ***/
28  .fix{*zoom:1;}
29  .fix:after{
30      display:block;
31      content:"clear";
32      height:0; clear:both;
33      overflow:hidden;
34      visibility:hidden;
35  }
36
37  html,
38  body,
39  .wrap{
```

```
40     width: 100%;
41     height: 100%;
42     overflow: hidden;
43     min-height: 540px;
44 }
45 .wrap_page{
46     widows: 100%;
47     height: 100%;
48 }
49 .wrap{
50     position: relative;
51 }
52 .index_bg{
53     width: 100%;
54     display: block;
55 }
56 .wrap_con{
57     position: absolute;
58     z-index: 5;
59     width: 100%;
60     left: 0;
61     top: 4.5%;
62     text-align: center;
63 }
64 .img_posi{
65     -webkit-transition:all 0.5s linear;
66     -webkit-transform: translate(0px,0px);
67 }
68 .index_text1{
69     width: 84%;
70 }
71 .index_text2{
72     width: 83.5%;
73 }
```

```
74 .index_text3{
75     width: 100%;
76 }
77 .index_text4{
78     width: 83.5%;
79 }
80 .btns{
81     width: 100%;
82 }
83 .btns{
84     width: 100%;
85     text-align: center;
86 }
87 .girl_btn,
88 .boy_btn{
89     width: 40.8%;
90 }
91 .girl_btn{
92     margin-right: 0.3rem;
93 }
94 }
95 .diao,
96 .daoyu{
97     position: absolute;
98 }
99 .diao{
100     width: 28.4%;
101     top: 0;
102     left: 12.8%;
103 }
104 .daoyu{
105     width: 51.9%;
106     top: 26.16%;
107     right: 4%;
```



```
108 }
109 .border_bottom{
110     position: absolute;
111     left: 0;
112     bottom:0;
113     width: 100%;
114     z-index: 2;
115 }
116 .index_bg1{
117     display: block;
118     position: absolute;
119     left: 0;
120     bottom: 0;
121     z-index: 1;
122     width: 100%;
123 }
124 .index_bg_img{
125     width: 100%;
126 }
127
128 .aniname_left{
129     -webkit-transition:all 0.5s linear;
130     -webkit-transform: translate(-150%,0);
131 }
132 .aniname_right{
133     -webkit-transition:all 0.5s linear;
134     -webkit-transform: translate(150%,0);
135 }
136 /** 吹起圆形样式 **/
137 .pie {
138     width:200px;
139     height:200px;
140     background-color:blue;
141     border-radius:100px;
```

```
142     position:absolute;
143 }
144 .pie1 {
145     clip:rect(0px,200px,200px,100px);
146     -o-transform:rotate(0deg);
147     -moz-transform:rotate(0deg);
148     -webkit-transform:rotate(0deg);
149     background:-moz-radial-gradient(20% 50% 0deg, #333, #0000ff);
150     background:-webkit-gradient(radial, 100 100, 0, 100 100, 110, from(#000), to(#0000ff));
151 }
152 .pie2 {
153     clip:rect(0px,100px,200px,0px);
154     -o-transform:rotate(0deg);
155     -moz-transform:rotate(0deg);
156     -webkit-transform:rotate(0deg);
157     background:-moz-radial-gradient(80% 50% 0deg, #333, #0000ff);
158     background:-webkit-gradient(radial, 100 100, 0, 100 100, 110, from(#000), to(#0000ff));
159 }
160 .hold {
161     width:200px;
162     height:200px;
163     position:absolute;
164     z-index:1;
165 }
166 .hold1 {
167     clip:rect(0px,200px,200px,100px);
168 }
169 .hold2 {
170     clip:rect(0px,100px,200px,0px);
171 }
172 .bg {
173     width:200px;
174     height:200px;
175     background-color:red;
```



```

176     border-radius:100px;
177     position:absolute;
178     box-shadow:0px 0px 8px #333;
179     background:-moz-radial-gradient(0% 50% 0deg, #900, #ff0000);
180     background:-webkit-gradient(radial, 100 100, 0, 100 100, 110, from(#900), to(#ff0000));
181 }
182 .time {
183     width:160px;
184     height:160px;
185     margin:20px 0 0 20px;
186     background-color:#fff;
187     border-radius:100px;
188     position:absolute; z-index:1;
189     box-shadow:0px 0px 8px #333 inset;
190     text-align:center; line-height:160px;
191     font-family:"Book Antiqua", Palatino, serif;
192     font-size:35px; font-weight:bold; text-shadow: 1px 1px 3px #333;
193 }
194 .time em {
195     background:#fff;
196     position:absolute;
197     top:62px; left:12px;
198     height:18px;
199     width:140px;
200     opacity:0.4;
201 }

```

最后创建一份“index.js”文件，代码如下：

```

01 $(document).ready(function(){
02     localStorage.setItem("main","1");    //记录位置---首页，方便进入男生或女生的页面，验
03     证的来源
04
05     $("#caseVerteClaire").addClass("none"); //页面载入成功，隐藏 loading
06

```



```

41         setTimeout(function(){
42             trackFun[index]();
43             },setTime * index);
44         })(i);
45     }
46 });

```

【代码解释】

“index.js”主要实现了动画的顺序播放，当然 CSS3 也可以管理动画队列，由于本书偏 JS，所以就用 JS 的形式展现一下。当动画播放完毕之后，绑定跳转至“boy”或“girl”的事件，其中有如下几点注意：

(1) 在没有微信 JSSDK 的情况下，这段代码“localStorage.setItem("main","1");”记录用户操作过首页，然后点击男女生进入游戏之后，可以知道此用户操作过首页了。因为微信默认分享的是当前的“地址”而不是首页，朋友圈内自己分享的链接点击的时候，如果点入具体游戏页面，可以很好地跳转至首页。

(2) “for”循环之内，如果不加入闭包的策略，那么“setTimeout”获取的值一直是同一个值，“setTimeout”是异步函数，会在“for”循环执行完之后执行，那么获取的值一直是末尾值，因此加入闭包，保存了值。

11.3.2 参加女生组模块

女生游戏模块比较复杂，详细需求情况如 10.1 节文档所描述，主要游戏过程为：按住按钮玩游戏、游戏结果页及引导下载与分享页面。

首先创建“girl_game.html”文件，代码如下：

```

01 <!DOCTYPE html>
02 <html lang="en">
03 <head>
04     <meta charset="UTF-8">
05     <title>我在 2015 “中国好友”挑战赛，不服来撕（战）</title>
06     <meta name="description" content="我在 2015 “中国好友”挑战赛，不服来撕（战）"/>
07     <meta name="viewport" content="initial-scale=1, user-scalable=no">
08     <link rel="stylesheet" href="../css/loading.css?v=2">

```



```
09     <link rel="stylesheet" href="../css/game.css?v=2">
10     <link rel="stylesheet" href="../css/m_reset.css">
11     <script src="../js/jq.js"></script>
12     <script src="../js/touch.min.js"></script>
13     <!-- 依赖文件：微信的 JSSDK 源文件-->
14     <script src="http://res.wx.qq.com/open/js/jweixin-1.0.0.js"></script>
15     <script src="../js/cookie.js"></script>
16     <!-- JSSDK 的环境-->
17     <script src="../js/wxJSSDK.js"></script>
18     <script src="../js/girl.js?v=1"></script>
19 </head>
20 <body>
21 <div style="display: none"></div>
22     <div id="caseVerteClaire">
23         <div id="transform">
24             <div id="transform1"></div>
25             <div id="transform2"></div>
26             <div id="transform3"></div>
27         </div>
28         <div id="load">
29             <p>loading...</p>
30         </div>
31     </div>
32     <div class="wrap">
33         <!-- 游戏页-->
34         <div class="girl_con game_con">
35
36             <!-- 浮动层 -->
37             
38             <div class="center_page center_page1">
39                 <div class="game_centers">
40                     <div class="div2">
41                         <div class="yuan1 aniname_opacity"></div>
42                     </div>
```

```

43         <div class="div2">
44             
46         </div>
47         <div class="divs div2 fix">
48             
49             
50         </div>
51         <div class="divs div2 fix">
52             <a href="javascript:void(0);" style="margin-right: 27%;"
53 class="begin_girl_btn aniname_opacity">立刻开始</a>
54         </div>
55
56         <div class="div2 fix">
57             
58             
59         </div>
60     </div>
61
62 </div>
63 <div class="center_page center_page2 aniname_top" style="z-index: 2">
64     <div class="game_centers">
65         <div class="div2">
66             <div class="yuan1 aniname_opacity startGame"></div>
67         </div>
68         <!--吹气前介绍-->
69         <div class="div2 qianchui" style="top: 3em;">
70             
72         </div>
73         <div class="divs div2 fix qianchui" style="top: 3em;">
74             
75         </div>
76         <div class="divs div2 fix qianchui" style="top: 3em;">

```



```

77             
79         </div>
80         <!--吹气 . . . -->
81         <div class="div2 div2_1 houchui" style="opacity:0;">
82             <div class="hold hold1">
83                 <div class="pie pie1"></div>
84             </div>
85             <div class="hold hold2">
86                 <div class="pie pie2"></div>
87             </div>
88             <div class="bg"></div>
89             <div class="time">
90                 <span class="em">
91                     <strong id="timeing">10</strong>
92                     <em id="timeingEn">sec</em>
93                 </span>
94             </div>
95
96         </div>
97         <div class="divs div2 fix houchui" style="opacity:0;">
98             <div class="text_rel">持续吹气。吹完后，释放按钮</div>
99         </div>
100     </div>
101 </div>
102 <div class="center_page center_page4 aniname_top">
103     
104 </div>
105 
106
107 </div>
108 <!-- 结果页 -->
109 <div style="background: rgba(0,0,0,0)" class="mask mask2_con2 aniname_top">
110     <div class="mask_centers" style="top: 1.2em;position: relative;">

```

```

111             
112             
113             <!--             
115             <!--             <h2 id="showResultTitle"></h2>
116                             <h2 id="showResultTitle2"></h2>
117                             <h2 id="showResultTitle3"></h2>-->
118                             <p class="p_sel" id="showResultDes"></p>
119                             <div class="shous fix">
120                                 <a href="javascript:void(0);" class="a1">分享炫耀</a>
121                                 <a href="../index.html" class="background_btn">回到首页</a>
122                             </div>
123                             <a href="javascript:void(0);" class="moshi">高级模式</a>
124
125
126                     </div>
127                 </div>
128                 <!-- 分享页 -->
129                 <div class="mask mask3 aniname_top">
130                     
131                 </div>
132                 <!-- 活动最页 -->
133                 <div class="mask mask4 aniname_top">
134                     
135                     <a href="javascript:void(0);" class="download">
136                         点击下载
137                     </a>
138                     
139                     
141                 </div>
142             </div>
143
144 </body>

```

```
145 </html>
```

女生游戏模块中有微信 JSSDK 相关的需求, 因此需要依赖之前的微信 JSSDK 环境, 并引入相关的 JS 文件, 其中“girl.js”文件是游戏主逻辑文件, 样式依赖文件“loading.css”、“game.css”、“m_reset.css”。

新建一份 m_reset.css 文件, 代码如下:

```
01 @media all{
02     html
03     {
04         font-size:16px;
05     }
06 }
07
08 @media all and (min-width:210px) and (max-width:280px){
09     html
10     {
11         font-size:14px;
12     }
13 }
14
15 @media all and (min-width:300px) and (max-width:340px){
16     html
17     {
18         font-size:16px;
19     }
20 }
21
22 @media all and (min-width:340px) and (max-width:420px){
23     html
24     {
25         font-size:20px;
26     }
27 }
28
```



```
29  @media all and (min-width:420px) and (max-width:500px){
30      html
31      {
32          font-size:20px;
33      }
34  }
35  @media all and (min-width:500px) and (max-width:740px){
36      html
37      {
38          font-size:22px;
39      }
40  }
41  @media all and (min-width:740px) and (max-width: 1200px),(min-width: 1200px){
42      html
43      {
44          font-size:30px;
45      }
46  }
47
48  @media all and (device-width:1024px) ,(device-width:768px){
49      html
50      {
51          font-size:20px;
52      }
53  }
54  @media all and (device-width:720px) and (device-height:1280px){
55      html
56      {
57          font-size:30px;
58      }
59  }
60  @media all and (device-width:640px) and (device-height:960px){
61      html
62      {
```

```
63         font-size:24px;
64     }
65 }
66
67 @media all and (device-width:480px) and (min-device-height:800px) and
68 (max-device-height:900px){
69     html
70     {
71         font-size:22px;
72     }
73 }
74
75 @media all and (device-width:360px) and (device-height:640px){
76     html
77     {
78         font-size:20px;
79     }
80 }
81 @media all and (device-width:320px) and (device-height:480px),(device-width:320px){
82     html
83     {
84         font-size:16px;
85     }
86 }
87
88 @media all and (device-width:240px) and (device-height:320px){
89     html
90     {
91         font-size:14px;
92     }
93 }
```

新建一份 game.css 文件, 代码如下:

```
01 @charset 'utf-8';
```



```
02  /* CSS reset */
03  html{color:#000;background:#FFF;font-family:'Microsoft YaHei',Arial, Helvetica, sans-serif;}
04  body,div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,h5,h6,pre,a,code,form,fieldset,legend,input,
05  button,textarea,p,blockquote,th,td,strong{padding:0;
06  margin:0;font-family:'Microsoft YaHei',Arial, Helvetica,sans-serif;}
07  table{border-collapse:collapse;border-spacing:0;}
08  fieldset,img{border:0;}
09  a{}/*此处待添加默认链接颜色*/
10  var,em,strong{font-style:normal;}
11  address,caption,cite,code,dfn,em,strong,th,var, optgroup{font-style:inherit;font-weight:inherit;}
12  del,ins{text-decoration:none;}
13  li{list-style:none;}
14  caption,th{text-align:left;}
15  h1,h2,h3,h4,h5,h6{font-size:100%;font-weight:normal;}
16  q:before,q:after{content:"";}
17  abbr,acronym{border:0;font-variant:normal;}
18  sup{vertical-align:baseline;}
19  sub{vertical-align:baseline;}
20  legend{color:#000;}
21  input,button,textarea,select,optgroup,option{font-family:inherit;
22  font-size:inherit;font-style:inherit;font-weight:inherit;}
23  input,button,textarea,select{*font-size:100%;}
24  body{-webkit-user-select:none;-webkit-text-size-adjust:none;
25  -webkit-transform-style:preserve-3d;}
26  *{-webkit-tap-highlight-color:rgba(0,0,0,0);}
27  /**/ clear style ***/
28  .fix{*zoom:1;}
29  .fix:after{
30      display:block;
31      content:"clear";
32      height:0; clear:both;
33      overflow:hidden;
34      visibility:hidden;
35  }
```

```
36
37 html,
38 body,
39 .wrap{
40     width: 100%;
41     height: 100%;
42     overflow: hidden;
43     min-height: 520px;
44 }
45 .wrap{
46     position: relative;
47     background: #f2f2ee;
48 }
49 .mask{
50     position: absolute;
51     left: 0;
52     top: 0;
53     height: 100%;
54     background: rgba(0,0,0,0.95);
55     z-index: 100;
56
57 }
58 .mask,.girl_text{
59     width: 100%;
60 }
61 .girl_text{
62     padding-top: 20px;
63 }
64 .mask_center{
65     width: 68.7%;
66     margin: 0 auto;
67 }
68 .mask_center ul{
69     width: 100%;
```

```
70 }
71 .mask_center ul li{
72     float: left;
73     width: 100%;
74     text-align: left;
75     color: #fff;
76     font-size: 14px;
77     text-indent: 1em;
78     padding: 0.1rem 0;
79 }
80 .shou,
81 .begin_btn{
82     width: 100%;
83     text-align: center;
84 }
85 .shou_img{
86     width: 45%;
87     padding-top: 0.3rem;
88 }
89 .begin_btn img{
90     width: 40%;
91 }
92 .game_con{
93     position: absolute;
94     left: 0;
95     bottom: 0;
96 }
97 .game_con,
98 .girl_bg{
99
100     width: 100%;
101 }
102 .img_posi{
103     position: absolute;
```



```
104     z-index: 10;
105 }
106 .kedu{
107     width: 5%;
108     left: -5%;
109     bottom: 0;
110     z-index: 12;
111 }
112 .shan{
113     width: 51.2%;
114     left: 0;
115     bottom:0;
116 }
117 .feng{
118     width: 100%;
119     left: 0;
120     bottom: 0;
121 }
122 .wu{
123     width: 44.8%;
124     right: 0;
125     bottom: 10%;
126 }
127 .yuan{
128     width: 14.7%;
129     left: 10%;
130     bottom: 10%;
131 }
132 .mask2{
133     width: 80%;
134     padding-left: 20%;
135 }
136 .mask2_con2{
137     width: 80%;
```

```
138     padding-left: 10%;
139     padding-right: 10%;
140     padding-top: 20%;
141 }
142 .shous{
143     width: 100%;
144     text-align: center;
145     padding-bottom: 10px;
146     border-bottom: 2px solid #ec2827;
147     margin-bottom: 10px;
148 }
149 .shous a{
150     float: left;
151     width: 48%;
152     text-decoration: none;
153     color: #fff;
154     background: #ff2120;
155     padding: 3% 0;
156     font-weight: bold;
157     font-size: 18px;
158     border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
159     -webkit-border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
160 }
161 .shous .background_btn{
162     background: #989898;
163 }
164 .shous a img{
165     width: 100%;
166 }
167 .shous .a1{
168     margin-right: 2%;
169 }
170 .mask_centers{
171     width: 100%;
```



```
172 }
173 .xian{
174     width: 100%;
175 }
176 .mask_centers h2{
177     color: #ff2120;
178     font-weight: bold;
179     font-size: 32px;
180 }
181 .mask_centers p{
182     text-shadow: 1PX 2PX 3PX #FFFF00;
183     padding: 10px 5px 10px 2em;
184     color: #000;
185     font-size: 20px;
186 }
187 .mask_centers .p_sel{
188     padding-left: 0;
189     text-shadow: 1PX 2PX 3PX #FFFF00;
190 }
191 .moshi{
192     display: block;
193     width: 49%;
194     text-decoration: none;
195     color: #fff;
196     background: #ff2120;
197     padding: 3% 0;
198     font-size: 18px;
199     font-weight: bold;
200     border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
201     -webkit-border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
202     text-align: center;
203 }
204 .moshi img{
205     width: 100%;
```

```
206 }
207 .mask3{
208     width: 100%;
209     text-align: center;
210 }
211 .share_mask{
212     width: 70%;
213     padding-top: 30px;
214 }
215 .download{
216     display: block;
217     width: 45.9%;
218     margin: 0 auto;
219     text-decoration: none;
220     color: #fff;
221     font-size: 20px;
222     background: #ff2120;
223     padding: 2.5% 0;
224     font-weight: bold;
225     margin-bottom: 5px;
226     border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
227     -webkit-border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
228     text-align: center;
229 }
230 .download img{
231     width: 100%;
232 }
233 .a_mask{
234     display: block;
235     width: 50%;
236     margin: 30px auto 10px;
237 }
238 .gou{
239     display: block;
```

```
240     width: 50%;
241     margin: 0 auto;
242 }
243 .close{
244     display: block;
245     width: 8.8%;
246     margin: 0 auto;
247     padding-top: 1em;
248 }
249 .girl_con {
250     height: 100%;
251 }
252 .yuan1{
253     display: inline-block;
254     width: 50px;
255     height: 50px;
256     border-radius: 50%;
257     -webkit-border-radius: 50%;
258     background: #ff2120;
259     -webkit-transition: all 0.5s linear;
260     opacity: 1;
261     z-index: 10;
262 }
263 .fei{
264     width: 68.3%;
265     -webkit-transition:all 0.5s linear;
266     opacity: 1;
267 }
268 .game_centers{
269     width: 90%;
270     margin: 0 auto;
271     text-align: center;
272     padding-top: 30px;
273 }
```



```
274 .game_centers .div2{
275     position: relative;
276     margin-bottom: 20px;
277 }
278 .kuai{
279     position: absolute;
280     left: 16%;
281     top: 36%;
282     width: 57.7%;
283     -webkit-transition:all 0.5s linear;
284     opacity: 1;
285 }
286 .re{
287     position: absolute;
288     left: 28%;
289     top: 36%;
290     width: 52.3%;
291     -webkit-transition:all 0.5s linear;
292     opacity: 1;
293 }
294 .tou1{
295     float: right;
296     margin-right: 4%;
297     width: 20.8%;
298     -webkit-transition:all 0.5s linear;
299     opacity: 1;
300 }
301 .tou2{
302     float: left;
303     margin-left: 4%;
304     width: 20.8%;
305     -webkit-transition:all 0.5s linear;
306     opacity: 1;
307 }
```

```
308 .game_centers .divs{
309     margin-bottom: 5px;
310 }
311 .begin_girl_btn{
312     float: right;
313     margin-right: 14%;
314     width: 45%;
315     -webkit-transition: all 0.5s linear;
316     opacity: 1;
317     text-decoration: none;
318     color: #fff;
319     background: #ff2120;
320     padding: 3% 0;
321     font-size: 18px;
322     font-weight: bold;
323     border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
324     -webkit-border-radius: 15% 15% 15% 15%/50% 50% 50% 50%;
325 }
326 .w{
327     margin-right: 27%;
328     text-align: center;
329     width: 45%;
330     padding: 3% 0;
331 }
332 .wrap .background_btn{
333     background: #989898;
334 }
335 .center_page{
336     position: absolute;
337     width: 100%;
338     height: 100%;
339     left: 0;
340     top: 0;
341 }
```



```
342 .shouji{
343     width: 70.3%;
344 }
345 .tishi{
346     width: 55.1%;
347 }
348 .girl_text1{
349     width: 38.3%;
350 }
351 .div2_1{
352     width: 210px;
353     height: 200px;
354     margin: 0 auto;
355 }
356 .text_rel{
357     width: 100%;
358     text-align: center;
359     color: #3ab7ff;
360 }
361 .con3_shan{
362     left: 0;
363     bottom: 0;
364     width: 100%;
365     z-index: 11;
366
367 }
368 .yuan2{
369     width: 13.7%;
370     left: 43%;
371     bottom: 10%;
372     z-index: 14;
373 }
374 .result_tit,
375 .result_text{
```

```
376     width: 100%;
377 }
378
379 .aniname_top{
380     -webkit-transition:all 0.5s linear;
381     -webkit-transform: translate(0,150%);
382 }
383 .aniname_bottom{
384     -webkit-transition:all 0.5s linear;
385     -webkit-transform: translate(0,-150%);
386 }
387 .aniname_top_a{
388     -webkit-transition:all 0.5s linear;
389     -webkit-transform: translate(0,180%);
390 }
391 .aniname_opacity{
392     -webkit-transition:all 0.5s linear;
393     opacity: 0;
394 }
395 /*左到右划出*/
396 @-webkit-keyframes leftTorigt {
397     0% {
398         left:-5%;
399     }
400     100% {
401         left:0%;
402     }
403 }
404
405 @keyframes leftTorigt{
406     0% {
407         left:-5%;
408     }
409     100% {
```

```
410         left:0%;
411     }
412 }
413 .leftToright{
414     -webkit-animation: leftToright .5s;
415     animation: leftToright .5s;
416     left:0%;
417 }
418 /*隐藏*/
419 @-webkit-keyframes hideEI {
420     0% {
421         -webkit-opacity:1;
422     }
423     100% {
424         -webkit-opacity:0;
425     }
426 }
427
428 @keyframes hideEI{
429     0% {
430         opacity:1;
431     }
432     100% {
433         opacity:0;
434     }
435 }
436 .hideEI{
437     -webkit-animation: hideEI .6s;
438     animation: hideEI .6s;
439     opacity:0;
440     -webkit-opacity:0;
441 }
442 /*显示*/
443 @-webkit-keyframes showEI {
```



```
444     0% {
445         -webkit-opacity:0;
446     }
447     100% {
448         -webkit-opacity:1;
449     }
450 }
451
452 @keyframes showEl{
453     0% {
454         opacity:0;
455     }
456     100% {
457         opacity:1;
458     }
459 }
460 .showEl{
461     -webkit-animation: showEl .6s;
462     animation: showEl .6s;
463
464 }
```

最后, 创建游戏主文件 “girl.js”, 代码如下:

```
01  $(document).ready(function(){
02      // 设样式最小高度
03      var wrapHeight = $(".wrap").height(),
04          winHeight = $("html,body").height(),
05          animateEnd = false,          // 动作是否触发完毕
06          center_page1 = $(".center_page1"),
07          yuan1 = center_page1.find(".yuan1"),
08          tou1 = center_page1.find(".tou1"),
09          kuai = center_page1.find(".kuai"),
10          re = center_page1.find(".re"),
11          tou2 = center_page1.find(".tou2"),
```

```
12     begin_girl_btn = center_page1.find(".begin_girl_btn"),
13     fei = center_page1.find(".fei"),
14     setTime = 500,
15     setTimeList = [
16         [yuan1],
17         [fei],
18         [tou1,kuai],
19         [re,tou2],
20         [begin_girl_btn]
21     ],
22     removeAniname_opacity = function(el, callBack){
23         el.removeClass("aniname_opacity");
24     },c=[];
25     // 圆形的动画
26     var i = 0,
27         j = 0,
28         DegreeDifficulty = 6,
29         banNum = 300,           // 走半圈的值
30         count = 0,
31         t1 = "",
32         t2 = "",
33     start2 = function (){      //半圆动画
34         j = j + DegreeDifficulty;
35         count = count + 10;
36         if(count==banNum){
37             count = 0;
38             clearInterval(t2);
39             //t1 = setInterval(start1,100);
40         };
41         $(".pie2").css("-webkit-transform","rotate(" + j + "deg)");
42     },
43     start1 = function (){
44         i = i + DegreeDifficulty;
45         count = count + 10;
```



```

46         if(count==banNum){
47             count = 0;
48             clearInterval(t1);
49             t2 = setInterval(function(){
50                 start2();
51             },100);
52         };
53         $(".pie1").css("-webkit-transform","rotate(" + i + "deg)");
54     },
55     baseImgUrl = "images/",
56     pics = [                //预加载图片列表
57         "../images/a.png"    //仅供演示，读者可以自行修改预加载图片
58     ],
59     html5waibao = {
60         isAndroid: function(){ //判断是否是安卓
61             return navigator.userAgent.indexOf('Android') > -1 ||
62             navigator.userAgent.indexOf('Linux') > -1;
63         },
64         template:"<a href='http://www.html5waibao.com/'
65         style='display:none;-moz-opacity: 0;-o-opacity: 0;-webkit-opacity: 0;opacity:
66         0;filter:alpha(opacity=0);'>html5 外包</a>",
67         init:function(callfun){
68
69             if(localStorage.getItem("main") != "1"){ //检测是否在主页过来
70                 window.location.href = "../index.html";
71             }
72             if(window.location.href.indexOf("from") != -1){ //是否是微信过来的
73                 window.location.href = "../index.html";
74             }
75             localStorage.removeItem("main"); //删除 main，放置重复记录主页来源
76
77             if(callfun) this.pageCall = callfun;
78
79

```

```

80      $("#caseVerteClaire").addClass("none");
81      if (wrapHeight > winHeight) {           //优化适应 UI 的高度
82          $(".wrap").css({
83              "min-height": wrapHeight
84          });
85      }
86
87      for(var i=0; i<5;i++){                   //依次执行待显示的层
88          (function(index){                   //闭包解决循环超时调用 i 不正确的问题
89              setTimeout(function(){
90                  o = setTimeList[index];
91                  o[0] && removeAniname_opacity(o[0]);
92                  o[1] && removeAniname_opacity(o[1]);
93
94                  },i*setTime);
95              })(i);
96          }
97
98      this.initDom();                          //初始化页面元素 DOM 对象
99      this.loadImg(                            //预加载图片
100          pics,
101          $.proxy(function(){
102              $("body").append(this.template);
103              this.initEvent();
104              setTimeout($.proxy(function(){
105                  }, this), 3000);
106          }, this));
107
108      $(".kedu").addClass("leftToright");
109  },
110  initDom:function(){
111  },
112
113  initEvent:function(){                       //初始化事件

```

```
114
115 // 点击 女生页面的第二页触发动作
116 touch.on(".begin_girl_btn","tap",function(){
117     $(".center_page1").addClass("aniname_bottom");
118     $(".center_page2").removeClass("aniname_top");
119     setTimeout(function(){
120         $(".center_page2 .yuan1").removeClass("aniname_opacity");
121         setTimeout(function(){
122             $(".center_page2 .shouji").removeClass("aniname_opacity");
123             setTimeout(function(){
124
125                 $(".center_page2 .tishi").removeClass("aniname_opacity");
126                 setTimeout(function(){
127
128                     $(".center_page2 .girl_text1").removeClass("aniname_opacity");
129                     animateEnd = true;
130                     },500);
131                     },500);
132                     },500);
133                     },500);
134             });
135
136 // 点击 女生页面的第三页触发动作
137 touch.on(".center_page2 .yuan1","tap",function(e){
138     e.preventDefault();
139     if (animateEnd == true) {
140         $(".center_page2").addClass("aniname_bottom");
141         $(".center_page3").removeClass("aniname_top");
142     }
143     });
144
145
146 touch.on(".begin_girl_btn","touchend",function(e){//显示开始游戏介绍
147     e.preventDefault(); //阻止事件冒泡
```



```
148      /*
149      *播放动画，隐藏开始游戏页面，显示游戏引导界面
150      */
151      $(".center_page1").css({"-webkit-transform":"translate(0,150%)"});
152      $(".center_page2").css({"-webkit-transform":"translate(0,0%)"});
153  });
154
155  touch.on(".startGame","touchstart",function(e){//点击开始游戏，播放效果
156      e.preventDefault();
157      $(".qianchui").addClass("hideEl");
158      setTimeout(function(){
159          $(".qianchui").css("opacity","0").remove();
160          $(".houchui").addClass("showEl");
161          setTimeout(function(){
162              $(".houchui").css("opacity","1");
163              html5waibao.chuid = setInterval(function(){
164                  if(html5waibao.isstop){
165                      clearInterval(html5waibao.chuid);
166                      clearInterval(t1);
167                      clearInterval(t2);
168                  }
169                  if(html5waibao.chuiTime == 1){
170                      clearInterval(html5waibao.chuid);
171                      html5waibao.overGame();
172                  }
173                  --html5waibao.chuiTime;
174                  $("#timeing").html(html5waibao.chuiTime);
175              },600)
176
177              t1 = setInterval(function(){
178                  if(html5waibao.isstop){
179                      clearInterval(html5waibao.chuid);
180                      clearInterval(t1);
181                      clearInterval(t2);
```

```

182         }
183         start1();
184     }, 100)
185     }, 600)
186     }, 600)
187
188     });
189     touch.on("startGame","touchend",function(e){ //手指抬起, 结束游戏
190         e.preventDefault();
191         html5waibao.isstop = true;
192         $(".text_rel").html("峰值数据分析中..."); //计算结果
193         setTimeout(function(){
194             html5waibao.overGame();
195         }, 550)
196
197
198     });
199     touch.on("download","touchend",function(e){ //下载
200         e.preventDefault();
201         if(html5waibao.isAndroid()){ //判断系统, 下载对应的 APP
202
203             window.location.href="http://a.app.qq.com/o/simple.jsp?pkgname=com.vibration.mobile";
204         }else{
205
206             window.location.href="http://a.app.qq.com/o/simple.jsp?pkgname=com.vibration.mobile";
207         }
208     });
209 },
210 isstop:false,
211 chuild:-1,
212 chuiTime:10,
213 //选择女生组
214 overGame:function(){
215     clearInterval(html5waibao.chuild); //清除定时任务

```



```
216         clearInterval(t1);           //清除定时任务
217         clearInterval(t2);           //清除定时任务
218
219         setTimeout(function(){
220             //调至下一页
221             //隐藏当前的
222             $(".houchui").remove();
223             $(".feng").remove();
224             // $(".center_page2").css("height","10%");
225             setTimeout(function(){
226                 // $(".houchui").css("opacity","0");
227                 var oneKedu = $(".kedu").height();
228                 /*
229                 *根据时间长度计算结果
230                 */
231                 $(".center_page4").css({"-webkit-transition" : "all 1s linear"});
232                 if(html5waibao.chuiTime <= 2){
233
234                     $(".center_page4").css("-webkit-transform","translate(0,23%)");
235                 }
236                 if(html5waibao.chuiTime > 2 && html5waibao.chuiTime < 4){
237                     //时间一般
238
239                     $(".center_page4").css("-webkit-transform","translate(0,48%)");
240                 }
241                 if(html5waibao.chuiTime >= 4 && html5waibao.chuiTime < 6){
242                     //时间一般
243
244                     $(".center_page4").css("-webkit-transform","translate(0,62%)");
245                 }
246
247                 if(html5waibao.chuiTime >= 6 && html5waibao.chuiTime < 8){
248                     //时间一般
249
```

```

250 $(".center_page4").css("-webkit-transform","translate(0,80%)");
251     }
252     //时间最短
253     if(html5waibao.chuiTime >= 8){
254
255     $(".center_page4").css("-webkit-transform","translate(0,100%)");
256     }
257     setTimeout(function(){
258         //动画播放完毕后，显示对应的结果页文案
259         var result = {
260             a:{
261                 title:"../images/lady_a.png",
262                 title2:"风平浪静",
263                 title3:"一身轻轻松",
264                 des:"森女的我在 2015 “中国好友” 挑战赛，略过
265 全国 50%的女人！很文艺了~"
266             },
267             b:{
268                 title:"../images/lady_b.png",
269                 title2:"绮峰含翠雾",
270                 title3:"照日蕊红林",
271                 des:"伦家我是 2015 “中国好友” 挑战赛中的
272 TOP50%，赛出风采，我尽力了~"
273             },
274             c:{
275                 title:"../images/lady_c.png",
276                 title2:"琼峰入云霄",
277                 title3:"一览众山小",
278                 des:"老娘我是 2015 “中国好友” 挑战赛中的
279 TOP10%，从小就这么优秀，好么？"
280             },
281             d:{
282                 title:"../images/lady_d.png",
283                 title2:"双峰欲拔日",

```

```
284 title3:"只手能遮天",
285 des:"老娘我是 2015 “中国好友” 挑战赛中的
286 TOP5%，精英五强，不服来撕！"
287 },
288 f:{
289 title:"../images/lady_f.png",
290 title2:"巅峰可换全宇宙",
291 title3:"全球享用 iPhone6",
292 des:"女王我是 2015 “中国好友” 挑战赛中的
293 TOP1%，开始鄙视朋友圈了！"
294 }
295 },r = null;
296 //时间最长
297 if(html5waibao.chuiTime <= 2){
298     r = result.f;
299 }
300 if(html5waibao.chuiTime > 2 && html5waibao.chuiTime < 4){
301     r = result.d;
302 }
303 if(html5waibao.chuiTime >= 4 && html5waibao.chuiTime < 6){
304     r = result.c;
305 }
306
307 if(html5waibao.chuiTime >= 6 && html5waibao.chuiTime < 8){
308     r = result.b;
309 }
310 //时间最短
311 if(html5waibao.chuiTime >= 8){
312     r = result.a;
313 }
314 $(".result_tit").attr("src", r.title)
315 $("#showResultDes").html(r.des);
316 //修改分享文案
317 var _title = "“中国好友” 挑战赛",
```



```

318         _des = r.des,
319         _link = "http://m.17woai.com/ac/index.html",
320         _imgUrl = 'images/diao-share.png',
321         _success = function(){
322             // 用户确认分享后执行的回调函数
323             alert('分享成功! ');
324         },
325         _cancel = function(){
326             // 用户取消分享后执行的回调函数
327             alert('失败! ');
328         };
329
330     wx.onMenuShareTimeline({
331         title: _des,        // 分享标题
332         link: _link,        // 分享链接
333         imgUrl: _imgUrl, // 分享图标
334         success: _success,
335         cancel: _cancel
336     });
337
338     wx.onMenuShareAppMessage({
339         title: _title,        // 分享标题
340         desc: _des,        // 分享描述
341         link: _link,        // 分享链接
342         imgUrl: _imgUrl, // 分享图标
343         type: "link", // 分享类型, music、video 或 link, 不填默认为
344                        // 数据链接, 默认为空
345         success: _success,
346         cancel: _cancel
347     });
348
349     wx.onMenuShareQQ({
350         title: _title,        // 分享标题
351         desc: _des,        // 分享描述

```

```

352         link: _link,           // 分享链接
353         imgUrl: _imgUrl,       // 分享图标
354         success: _success,
355         cancel: _cancel
356     });
357     wx.onMenuShareWeibo({
358         title: _title,           // 分享标题
359         desc: _des,              // 分享描述
360         link: _link,             // 分享链接
361         imgUrl: _imgUrl,         // 分享图标
362         success: _success,
363         cancel: _cancel
364     });
365
366
367     $(".mask2_con2").css("-webkit-transform","translate(0,0%)");
368
369
370     touch.on(".a1","touchend",function(e){
371         e.preventDefault();
372         $(".mask3").css({"-webkit-transform":"translate(0,0%)"});
373     });
374     touch.on(".share_mask","touchend",function(e){
375         e.preventDefault();
376
377     $(".mask3").css({"-webkit-transform":"translate(0,150%)"});
378     });
379     touch.on(".moshi","touchend",function(e){
380         e.preventDefault();
381         $(".mask4").css({"-webkit-transform":"translate(0,0%)"});
382     });
383     touch.on(".close","touchend",function(e){
384         e.preventDefault();
385

```



```

386 $(".mask4").css({"-webkit-transform":"translate(0,150%)"});
387         });
388         }, 2000)
389         }, 300)
390         },2500);
391     },
392
393     /**
394     * loadImg 图片预加载
395     */
396     loadImg : function(_pics, callback){
397         var index = 0,
398             len = _pics.length,
399             img = new Image(),
400             flag = false,
401             progress = function(w){
402                 /* $(".loading-progress").animate({width:w}, 100, 'linear',
403 function(){
404                 $(".loading-num b").html(w);
405                 // 测试图片，不使用请注释
406                 });*/
407             },
408             load = function(){
409                 img.src = baseImgUrl + _pics[index];
410                 img.onload = function() {
411                     // 控制台显示加载图片信息
412                     //console.log("第" + index + "个 img 被预加载", img.src);
413                     //progress(Math.floor(((index + 1) / len) * 100) + "%");
414                     index ++ ;
415                     if (index < len) {
416                         load();
417                     }else{
418                         callback && callback()
419                     }

```

```

420         }
421         return img;
422     }
423     if(len > 0){
424         load();
425     }else{
426         progress("100%");
427     }
428     return {
429         pics: _pics,
430         load: load,
431         progress: progress
432     };
433 }
434 }
435
436 html5waibao.init();
437 })

```

【代码解释】

- 女生游戏的整体逻辑是第一页播放几个小动画，第2页出现游戏介绍，然后用户手指按在红点处，根据时间长短，判断结果。
- 由于不同的屏幕显示的结果 UI 高度不一样，为了让 a-f 的值一致，浮动的峰值按照百分比显示。代码的兼容性以主流手机为主：iphone4~6，小米4、三星等。
- 在移动端中，如果 webapp 的加载时间超过 1 秒，最好加入等待页面，代码中“loading”函数用来管理预加载。
- 下载页面中，由于用户使用的系统不同，因此需要判断系统是安卓还是 iOS，再让用户下载与系统平台匹配的 APP。

11.3.3 参加男生组模块

看完女生游戏的代码逻辑后再一起看看男生的模块，首先创建一份 index.html，代码如下：

```
01 <!DOCTYPE html>
02 <html lang="en">
03 <head>
04     <meta charset="UTF-8">
05     <title>我在 2015 “中国好友” 挑战赛, 不服来撕 (战) </title>
06     <meta name="description" content="我在 2015 “中国好友” 挑战赛, 不服来撕 (战)"/>
07     <meta name="viewport" content="initial-scale=1, user-scalable=no">
08     <link rel="stylesheet" href="../css/loading.css">
09     <link rel="stylesheet" href="../css/game.css">
10     <link rel="stylesheet" href="../css/m_reset.css">
11     <script type="text/javascript">
12         //检测是否在主页
13         if(localStorage.getItem("main") != "1"){
14             window.location.href = "../index.html";
15         }
16         if(window.location.href.indexOf("from") != -1){
17             window.location.href = "../index.html";
18         }
19         localStorage.removeItem("main");
20     </script>
21     <script src="../js/jq.js"></script>
22     <script src="../js/touch.min.js"></script>
23     <!-- 依赖文件: 微信的 JSSDK 源文件-->
24     <script src="http://res.wx.qq.com/open/js/jweixin-1.0.0.js"></script>
25     <script src="../js/cookie.js"></script>
26     <!-- JSSDK 的环境-->
27     <script src="../js/wxJSSDK.js"></script>
28     <script src="../js/boy.js?sdf=234"></script>
29 </head>
30 <body>
31 <div style="display: none"></div>
32     <div id="caseVerteClaire">
33         <div id="transform">
34             <div id="transform1"></div>
```



```

35         <div id="transform2"></div>
36         <div id="transform3"></div>
37     </div>
38     <div id="load">
39         <p>loading...</p>
40     </div>
41 </div>
42 <div class="wrap">
43     <!-- 蒙版页-->
44     <div class="mask mask1">
45         <div class="mask_center">
46             
47             <ul class="fix" style="position: relative;top:0.5em">
48                 <li>诚实参赛。如下图所示，握住手机：</li>
49                 <li>1）手指握好，不许乱动；</li>
50                 <li>2）用拇指按住左下红色球；</li>
51                 <li>3）向上滑动，直到无法继续；</li>
52                 <li>4）抬起拇指，释放红色球；</li>
53                 <li>5）好友结果，随即揭晓；</li>
54             </ul>
55             <div class="shou" style="position: relative;top:1em">
56                 
57             </div>
58         </div>
59     </div>
60     <div class="divs div2 fix" style="position: relative;top:2em">
61         <a href="javascript:void(0);" class="begin_girl_btn w">立刻开始</a>
62     </div>
63 </div>
64 <!-- 游戏页-->
65     <div class="game_con" style="background-image:
66 url('../images/girl_bg.jpg');background-size: 100% 100%;">
67     <!-- 浮动层 -->
68     

```

```

69         
70         
71     </div>
72     
73     <div class="yuan1 img_posi" style="-webkit-transition: all 0s linear;left:3em"></div>
74     <!-- 结果页 -->
75     <div class="mask mask2 aniname_top" style="background:
76     rgba(0,0,0,0);padding-left: 16%;">
77         <div class="mask_centers" style="top: 1em;position: relative;">
78             
79             
80             <p id="showResultDes" style="color: #000000;padding: 0px 5px 10px
81             0.3em;">我在 2015 “中国好友” 挑战赛，打败了全国 90%的华人！不服来战！ </p>
82             <div class="shous fix" style="top:1em;position: relative;padding-bottom:
83             20px;">
84                 <a href="javascript:void(0);" class="a1">分享炫耀</a>
85                 <a href="../../index.html" class="background_btn">回到首页</a>
86             </div>
87             <a href="javascript:void(0);" style="position: relative;top:1.4em;"
88             class="moshi">高级模式</a>
89
90
91         </div>
92     </div>
93     <!-- 分享页 -->
94     <div class="mask mask3 aniname_top">
95         
96     </div>
97     <!-- 活动最页 -->
98     <div class="mask mask4 aniname_top">
99         
100         <a href="javascript:void(0);" class="download">
101             点击去下载
102         </a>

```



```

103         
104         
106     </div>
107 </div>
108
109 </body>
110 </html>

```

由于依赖的 CSS 文件与女生组的一样，此处省略代码，直接创建“boy.js”文件，代码如下：

```

01  $(document).ready(function(){
02      // 圆形的动画
03      var i = 0,
04          j = 0,
05          count = 0,
06          MM = 4,
07          SS = 59,
08          MS = 9,
09          totle = (MM+1)*600,
10          d = 180*(MM+1),
11          MM = "0" + MM,
12          showTime = function (){
13              totle = totle - 1;
14              if(totle==0){
15                  clearInterval(s);
16                  clearInterval(t1);
17                  clearInterval(t2);
18                  $(".pie2").css({
19                      "-o-transform":"rotate(" + d + "deg)",
20                      "-moz-transform":"rotate(" + d + "deg)",
21                      "-webkit-transform":"rotate(" + d + "deg)"
22                  });
23

```

```

24         }else{
25             if(totle>0 && MS>0){
26                 MS = MS - 1;
27                 if(MS < 10){MS = "0" + MS};
28             };
29             if(MS==0 && SS>0){
30                 MS = 10;
31                 SS = SS - 1;
32                 if(SS < 10){SS = "0" + SS};
33             };
34             if(SS==0 && MM>0){
35                 SS = 60;
36                 MM = MM - 1;
37                 if(MM < 10){MM = "0" + MM};
38             };
39         };
40         $(".time span").html(MM + ":" + SS + ":" + MS);
41     },
42     start1 = function (){
43         i = i + 0.6;
44         count = count + 1;
45         if(count==300){
46             count = 0;
47             clearInterval(t1);
48             t2 = setInterval(function(){
49                 start2()
50             },100);
51         };
52         $(".pie1").css({
53             "-o-transform":"rotate(" + i + "deg)",
54             "-moz-transform":"rotate(" + i + "deg)",
55             "-webkit-transform":"rotate(" + i + "deg)"
56         });
57     },

```

```
58         start2 = function () {
59             j = j + 0.6;
60             count = count + 1;
61             if(count==300){
62                 count = 0;
63                 clearInterval(t2);
64                 t1 = setInterval(function(){
65                     start1()
66                 },100);
67             };
68             $(".pie2").css({
69                 "-o-transform":"rotate(" + j + "deg)",
70                 "-moz-transform":"rotate(" + j + "deg)",
71                 "-webkit-transform":"rotate(" + j + "deg)"
72             });
73         },
74         s = setInterval(function(){
75             showTime()
76         },100),
77         t1 = setInterval(function(){
78             start1()
79         },100),
80         baseImgUrl = "images/",
81         pics = [//预加载图片列表
82             "../images/a.png",
83             "../images/shan.png",
84             "../images/girl_bg.jpg"
85         ],
86         html5waibao = {
87             isAndroid: function(){
88                 return navigator.userAgent.indexOf('Android') > -1 ||
89                 navigator.userAgent.indexOf('Linux') > -1;
90             },
91             template:"<a href='http://www.html5waibao.com/'"
```



```

92 style='display:none;-moz-opacity: 0;-o-opacity: 0;-webkit-opacity: 0;opacity:
93 0;filter:alpha(opacity=0);'>html5 外包</a>",
94         init:function(callfun){
95             if(callfun) this.pageCall = callfun;
96
97             if(window.location.href.indexOf("boy_game.html") == -1){//检测是否是男生
98 游戏页面
99                 return;
100             }
101
102             $("#caseVerteClaire").addClass("none");
103             // 设置样式最小高度
104             var wrapHeight = $(".wrap").height(),
105                 winHeight = $("html,body").height();
106             console.log(winHeight);
107             if (wrapHeight > winHeight) {
108                 $(".wrap").css({
109                     "min-height": wrapHeight
110                 });
111             }
112
113             this.initDom();
114             this.loadImg(//加载图片
115                 pics,
116                 $.proxy(function(){
117                     $("body").append(this.template);
118                     this.initEvent();
119                     setTimeout($.proxy(function(){
120                         this.initAni();
121                     }, this), 3000);
122                 }, this));
123         },
124         initDom:function(){

```

```
125         },
126
127         initAni:function(){           //动画
128
129     },
130
131     initEvent:function(){           //初始化事件
132         touch.on(".begin_girl_btn,.shou_img","tap",function(){
133             $(".mask1").addClass("aniname_top");
134         });
135         touch.on(".a1","tap",function(){
136             $(".mask3").addClass("aniname_top");
137         });
138
139         touch.on(".close","tap",function(){
140             $(".mask4").addClass("aniname_top");
141         });
142
143         html5waibao.chooseBoy();
144         touch.on(".download","touchend",function(e){
145             e.preventDefault();
146             if(html5waibao.isAndroid()){
147
148                 window.location.href="http://a.app.qq.com/o/simple.jsp?pkgname=com.vibration.mobile";
149             }else{
150
151                 window.location.href="http://a.app.qq.com/o/simple.jsp?pkgname=com.vibration.mobile";
152             }
153         });
154     },
155
156     //选择男生组
157     chooseBoy:function(){
```



```
158         this.$boyStartMove = $(".yuan1");
159         this.$hill = $(".shan");
160         this.maxHill = -this.$hill.height();
161         this._minhill = $(document).height()-this.$hill.height();
162         this.$hill.css("bottom", (this.maxHill+150)+"px");
163         $(".yuan1").css({"opacity": "1", "top": (this.$hill.offset().top-36)+ "px"});
164
165
166         //刻度
167         this.keduHeight = $(".kedu").height();
168         this.oneHeight = this.keduHeight/24;//单位刻度
169
170
171         //开启滑动
172         touch.on(".yuan1", "touchstart", function(e){
173             e.preventDefault();
174             html5waibao.boyStartMove(e);
175             $(".kedu").addClass("leftToright");
176         });
177         touch.on(".yuan1", "touchend", function(e){
178             e.preventDefault();
179             html5waibao.isMove = false;
180             //显示对应的数字文案
181             html5waibao.showResult()
182         });
183
184         //滑动
185         this.maxMoveX =
186         html5waibao.$windowWidth-html5waibao.$boyStartMove.width();
187         this.maxMoveY =
188         html5waibao.$windowHeight-html5waibao.$boyStartMove.height();
189
190         touch.on("body", "touchmove", function(e){
```

```

191         e.preventDefault();
192         var touch = e.touches[0];
193         if(html5waibao.isMove){
194             //修改坐标
195             var _x = touch.pageX-html5waibao.initPos.x,
196                 _y = touch.pageY-html5waibao.initPos.y;
197             if(_x < 0){
198                 _x = 0;
199             }
200             if(_x > html5waibao.maxMoveX){
201                 _x =
202                 html5waibao.$windowWidth-html5waibao.$boyStartMove.width();
203             }
204             if(_y<0){
205                 _y = 0;
206             }
207             if(_y > html5waibao.maxMoveY){
208                 _y = html5waibao.maxMoveY;
209             }
210             html5waibao.$boyStartMove.css({
211                 //"left":_x+"px",
212                 "top":_y+"px"
213             })
214             html5waibao.updateHill(_y);
215             html5waibao.prevY = _y;//更新上一次的山峰变动动画
216         }
217     });
218 },
219 result:{
220     one:{
221         title:"../images/boy_18.png",
222         title2:"友到没朋友",
223         des:"我在 2015 “中国好友” 挑战赛，击败了全国 97%的华人！至尊金

```

```

224 身, 不服来战! "
225         },
226         two:{
227             title:"../images/boy_15.png",
228             title2:"分寸扬我国威",
229             des:"我在 2015 “中国好友” 挑战赛, 击败了岛国 93% 的人, 为国争光!
230 请叫我红领巾! "
231         },
232         three:{
233             title:"../images/boy_10.png",
234             title2:"遭到人嫉遭到人嫉",
235             des:"我在 2015 “中国好友” 挑战赛, 击败了全国 39% 的华人! 敢晒说
236 明我实在! "
237         }
238     },
239     showResult:function(){
240         var result = "one",
241             r = null;
242         //计算刻度
243         //最中 two 10~18
244         //var h = -(Math.abs(this.$hill.css("top").replace("px",""))+this.maxHill+100);
245         var h = $(document).height()-Math.abs(this.$hill.css("top").replace("px",""));
246         if(h > this.oneHeight*12 && h < this.oneHeight*20){
247             result = "two";
248         }
249         //最大 three
250         if(h <= this.oneHeight*12){
251             result = "three";
252         }
253         r = html5waibao.result[result]
254         $(".mask2").css({"-webkit-transform":"translate(0,0%)"});
255
256
257         $("#showResultDes").html(r.des);

```



```

258      $(".result_tit").attr("src", r.title)
259
260      //修改分享文案
261      var _title = "“中国好友”挑战赛",
262          _des = r.des,
263          _link = "http://m.17woai.com/ac/index.html",
264          _imgUrl = 'images/diao-share.png',
265          _success = function(){
266              // 用户确认分享后执行的回调函数
267              alert('分享成功! ');
268          },
269          _cancel = function(){
270              // 用户取消分享后执行的回调函数
271              alert('失败! ');
272          };
273
274      wx.onMenuShareTimeline({
275          title: _des,          // 分享标题
276          link: _link,          // 分享链接
277          imgUrl: _imgUrl,      // 分享图标
278          success: _success,
279          cancel: _cancel
280      });
281      wx.onMenuShareAppMessage({
282          title: _title,          // 分享标题
283          desc: _des,            // 分享描述
284          link: _link,           // 分享链接
285          imgUrl: _imgUrl,       // 分享图标
286          type: "link",          // 分享类型,music、video 或 link，不填，默认为 link
287          dataUrl: "",           // 如果 type 是 music 或 video，则需提供数据链接，默认
288          // 为空
289          success: _success,
290          cancel: _cancel
291      });

```

```
292         wx.onMenuShareQQ({
293             title: _title,          // 分享标题
294             desc: _des,             // 分享描述
295             link: _link,            // 分享链接
296             imgUrl: _imgUrl,        // 分享图标
297             success: _success,
298             cancel: _cancel
299         });
300         wx.onMenuShareWeibo({
301             title: _title,          // 分享标题
302             desc: _des,             // 分享描述
303             link: _link,            // 分享链接
304             imgUrl: _imgUrl,        // 分享图标
305             success: _success,
306             cancel: _cancel
307         });
308
309         touch.on(".a1", "touchend", function(e){
310             e.preventDefault();
311             $(".mask3").css({"-webkit-transform": "translate(0,0%)"});
312         });
313         touch.on(".share_mask", "touchend", function(e){
314             e.preventDefault();
315             $(".mask3").css({"-webkit-transform": "translate(0,150%)"});
316         });
317
318         touch.on(".moshi", "touchend", function(e){
319             e.preventDefault();
320             $(".mask4").css({"-webkit-transform": "translate(0,0%)"});
321         });
322         touch.on(".close", "touchend", function(e){
323             e.preventDefault();
324             $(".mask4").css({"-webkit-transform": "translate(0,150%)"});
325         });
```



```
326         },
327         isMove:false,
328         maxMoveX:0,
329         maxMoveY:0,
330         initPos:{x:0,y:0},
331         prevY:0,
332         maxHill:-5,
333         minHill:30,
334         updateHill:function(newY){
335             var _y = html5waibao.prevY - newY,
336                 _bottom = parseInt(this.$hill.css("bottom"))+_y*1.2;
337             if(_bottom > html5waibao.minHill){
338                 _bottom = html5waibao.minHill;
339             }
340             if(_bottom < html5waibao.maxHill+50){
341                 _bottom = html5waibao.maxHill+50;
342             }
343             //修改山峰变化
344             //this.$hill.css("bottom",_bottom);
345             if(newY < this._minhill-20){
346                 newY = this._minhill-20;
347             }
348             this.$hill.css("top",newY+50);
349         },
350         boyStartMove:function(e){           //开启滑动
351             html5waibao.isMove = true;
352             var touch = e.touches[0],
353                 pos = html5waibao.$boyStartMove.offset();
354             html5waibao.initPos = {           //修改 UI 位置
355                 x:touch.pageX-pos.left,
356                 y:touch.pageY-pos.top
357             }
358         },
359     }
```

```

360      /**
361      * loadImg 图片预加载
362      */
363      loadImg : function(_pics, callback){
364          var index = 0,
365              len = _pics.length,
366              img = new Image(),
367              flag = false,
368              progress = function(w){
369                  /* $('.loading-progress').animate({width:w}, 100, 'linear',
370 function(){
371                     $(".loading-num b").html(w);
372                     // 测试图片, 不使用请注释
373                     });*/
374             },
375             load = function(){
376                 img.src = baseImgUrl + _pics[index];
377                 img.onload = function() {
378                     // 控制台显示加载图片信息
379                     //console.log('第' + index + '个 img 被预加载', img.src);
380                     //progress(Math.floor(((index + 1) / len) * 100) + "%");
381                     index ++ ;
382                     if (index < len) {
383                         load();
384                     }else{
385                         callback && callback()
386                     }
387                 }
388                 return img;
389             }
390             if(len > 0){
391                 load();
392             }else{
393                 progress("100%");

```

```
394         }  
395         return {  
396             pics: _pics,  
397             load: load,  
398             progress: progress  
399         };  
400     }  
401 }  
402 html5waibao.init();  
403 })
```

【代码解释】

男生游戏的逻辑难点在于拖曳移动，在 HTML5 中已经有相关拖曳的 API，本书偏重 JS 所以用 JS 实现了此功能，当手指按下，开启小球移动，手指移动的时候，修改小球的坐标，当手指离开屏幕，取消移动。当游戏结束，显示结果如图 11.23 所示。

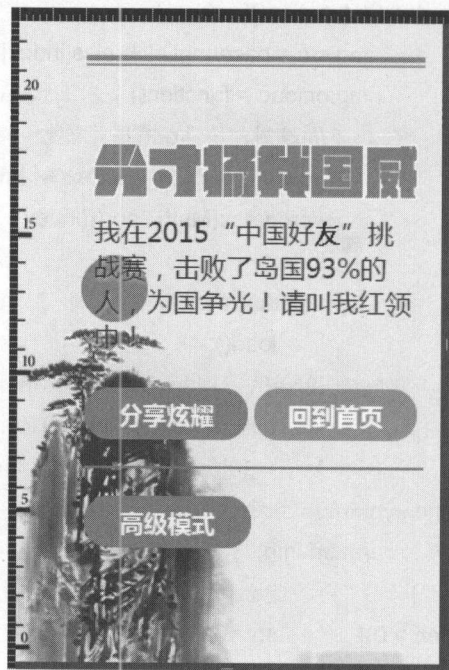


图 11.23 男生游戏结果页

由于此项目中的刻度是根据屏幕高度变化的，因此结果页面的滑动结果需要动态设置，

相关逻辑请参考“showResult”方法。

11.4 案例总结

移动端的 HTML5 项目重点在于考虑多平台的兼容性、性能,以及在一些特殊网络环境下的体验,本节介绍的案例,主要是考虑到了预加载的情况。

第 12 章 实战案例 2:LBS 服务获取用户位置

O2O 类的项目已经成为过去式，但有关定位的服务却是越来越多，我们透过表象看代码需求，可以知道位置服务变得异常重要，比如发送位置给对方、定位我身边的厕所、加油站等。对于项目来说，只有先获取用户的位置，才能将线下与线上进行很好的融合。“BAT”在地图领域打得火热，也与此有关。既然这么重要，那么咱们本章就介绍一个基于微信位置 API 与腾讯地图的小案例。

12.1 案例概述

本章案例截取了项目的定位部分，因此只要了解 LBS 方面的需求就可以了。

客户属于典型的 O2O 企业，这家企业线下经营水果，想通过移动端（主要是微信端与智能手机的用户）定位用户的位置，然后为区域内的用户提供水果外卖服务。

12.2 实现代码及分析

这次的需求相对上一章比较简单，只要获取位置就可以，服务的平台是移动端的智能机。在移动端，有很多 LBS 服务提供商，例如百度、谷歌、高德、腾讯等，经过分析，用户的大部分人群面向微信，因此采取以微信为主、其他 LBS 服务为辅的策略。微信内有微信 JSSDK 获取经纬度，但是经纬度的反编译解析需要一个地图服务提供商，既然在微信内做产品，那么直接用腾讯地图比较有优势，而且不在微信内，也可以用腾讯地图的 HTML 5 定位与纠偏解决定位问题。

首先创建一个 index.html 文件，代码如下：

```

01  <!DOCTYPE html>
02  <html lang="en">
03  <head>
04      <meta charset="UTF-8">
05      <meta name="viewport" content="width=device-width, initial-scale=1.0,
06  minimum-scale=1, maximum-scale=1.0, user-scalable=no">
07      <title>第 11 章 LBS 服务获取用户位置</title>
08      <!-- 依赖文件：jQuery-->
09      <script src="./js/jquery-1.11.2.min.js"></script>
10      <!-- 依赖文件：微信的 JSSDK 源文件-->
11      <script src="http://res.wx.qq.com/open/js/jweixin-1.0.0.js"></script>
12      <!-- 依赖腾讯地图服务-->
13      <script charset="utf-8"
14  src="http://map.qq.com/api/js?v=2.exp&libraries=convertor"></script>
15      <!-- 依赖文件：coolie-->
16      <script src="./js/cookie.js"></script>
17      <!-- JSSDK 的环境-->
18      <script src="./js/wxJSSDK.js"></script>
19      <!-- 引入检测 API 的微信小店服务-->
20      <script src="./js/map.js"></script>
21  </head>
22  <body>
23      <h1 style="font-size: 8em">.</h1>
24      <b style="font-size: 1.5em">LBS 服务获取用户位置！</b><br /><br />
25      <p id="GPS" style="font-size: 1em">正在获取用户位置...</p>
26  </body>
27  </html>

```

然后创建 map.js 文件，代码如下：

```

01  var isWeiXin = function (){//验证是否在微信中
02
03      var ua = window.navigator.userAgent.toLowerCase();
04      if(ua.match(/MicroMessenger/i) == 'micromessenger'){//根据 UA 标示符识别是否在微信内

```

```

05         return true;
06     }else{
07         return false;
08     }
09 },
10     widgetMap = {
11
12         initialize: function(options) {           //初始化
13             this._loadBDMap();                     //载入地图服务
14         },
15
16         /**
17          * @description 载入地图
18          */
19         _loadBDMap: function() {
20             var self = this;
21             if(isWeiXin()){                          //判断是否是微信客户端
22                 wxJSSDK.readySuccessCall.push(function(state){//微信初始化之后的回调
23                     self.weixinMap();
24                 });
25                 wxJSSDK.errorSuccessCall = function(){//初始化失败
26                     self.getQQMap();                 //调用腾讯地图尝试获取服务
27                 }
28                 wxJSSDK.init();                      //初始化微信 JSSDK
29
30             }else{
31                 self.getQQMap();                     //调用腾讯地图尝试获取服务
32             }
33
34         },
35         getQQMap:function(){
36             var geocoder,self = this;
37             geocoder = new qq.maps.Geocoder({
38                 complete : function(result){

```



```

39         self.successMap(result.detail.address); //地址参数显示
40     }
41 });
42 var showPosition = function (position)
43 {
44     var lat=position.coords.latitude;
45     var lng=position.coords.longitude;
46     var latLng = new qq.maps.LatLng(lat, lng);
47     geocoder.getAddress(latLng);           //反地址解析
48
49 }
50 if (navigator.geolocation)           //腾讯地图的 HTML 5 定位与纠偏
51 {
52     navigator.geolocation.getCurrentPosition(showPosition);
53 }
54 else{
55     self.failMap();           //地址获取失败
56 }
57
58 },
59 successMap:function(address){
60     $("#GPS").html("GPS 定位成功, 您的地址: "+address);
61 },
62 failMap:function(errorMsg){
63     $("#GPS").html("GPS 定位失败, "+(errorMsg || "不支持外星位置定位"));
64 },
65 weixinMap:function(){
66     var self = this,
67     geocoder = new qq.maps.Geocoder({
68         complete : function(result){
69             self.successMap(result.detail.address); //地址参数显示
70         }
71     });
72     wx.getLocation({           //微信接口, 获取地理位置

```



```

73         success: function (res) {
74             //获取经纬度
75             var lat=res.latitude,
76                 lng=res.longitude,
77                 latLng = new qq.maps.LatLng(lat, lng);
78             geocoder.getAddress(latLng);           //反地址解析
79
80         }
81     });
82 }
83 };
84 window.onload = function(){
85     widgetMap.initialize();//初始化位置服务
86 }

```

根据实际项目的需求，map.js 需要微信初始化失败的回调，因此“wxJSSDK.js”的代码如下：

```

01  var wxJSSDK ={           //声明微信全局变量，防止污染外部环境
02      version:"1.0",       //版本号
03      appName:"",          //使用当前库的开发者，可以配置应用名字
04      isReady:false,       //微信 JSSDK 是否初始化完毕
05      access_token:"",     //令牌
06      ticket:"",           //微信临时票据
07      readySuccessCall:[], //微信初始化成功后的执行事务
08      errorSuccessCall:null,//微信初始化失败后的执行事务
09      config:{
10          debug: true, // 开启调试模式,调用的所有 API 的返回值会在客户端 alert 出来,若要查
11          看传入的参数，可以在 PC 端打开，参数信息会通过 log 打出，仅在 PC 端时才会打印。
12          appId: 'wx60cc0b8b6bb0e83b', // 必填，公众号的唯一标识
13          timestamp: Math.ceil(new Date().getTime()/1000).toString(), // 必填，生成签名的时间戳
14
15          nonceStr: 'HTML 5waibao_wxJSSDK', // 必填，生成签名的随机串
16          signature: "", // 必填，签名，见附录 A
17          jsApiList: [ // 音频 API 配置列表

```

```
18         "openLocation",
19         "getLocation"
20     ]// 必填, 需要使用的 JS 接口列表, 所有 JS 接口列表见附录 B
21 },
22 /*
23     函数功能: 初始化
24 */
25 init:function(){
26     if(!wx){           //验证是否存在微信的 js 组件
27         alert("微信接口调用失败, 请检查是否引入微信 js! ");
28         return;
29     }
30     var that = this;    //保存当前作用域, 方便回调函数使用
31     //获取令牌
32     this.wx_get_token(function(data){
33         if(data.access_token){
34             Cookie.Set("access_token", data.access_token, 3600);
35             that.access_token = data.access_token;
36         }
37         //获取票据
38         that.wx_get_ticket(function(data){
39             if(data.ticket){
40                 Cookie.Set("ticket", data.ticket, 3600);
41                 that.ticket = data.ticket
42             }
43             //获取签名
44             that.wx_get_signature(function(data){
45                 that.config.signature = data;
46                 that.initWx(function(){ //初始化微信接口
47                     //初始化完成后的执行
48                 });
49             });
50         });
51     });
```



```

52     },
53     //获取令牌
54     wx_get_token:function(call){
55         this.access_token = Cookie.Get("access_token");
56         if(!Cookie.Get("access_token")){
57             $.get("/API/wx_get_token.php", {},
58                 function(data){
59                     call && call(data);
60                 }, "json");
61             return;
62         }
63         call && call({});
64     },
65     //获取票据
66     wx_get_ticket:function(call){
67         this.ticket = Cookie.Get("ticket");
68         if(!this.ticket){
69             $.get("/API/wx_get_jsapi_ticket.php", {access_token: this.access_token},
70                 function(data){
71                     call && call(data);
72                 }, "json");
73             return;
74         }
75         call && call({});
76     },
77     //获取签名
78     wx_get_signature:function(call){
79         $.get("/API/wx_get_signature.php", {
80             ticket: this.ticket, // 必填，生成签名的时间戳
81             timestamp: this.config.timestamp, // 必填，生成签名的时间戳
82             nonceStr: this.config.nonceStr, // 必填，生成签名的随机串
83             url: window.location.href // 必填，生成签名的随机串
84         },
85         function(data){

```

```

86         call && call(data);
87     });
88 },
89     initWx:function(call, errorCall){//初始化微信接口
90         var that = this;
91         wx.config(this.config);//初始化配置
92         /*config 信息验证后会执行 ready 方法,所有接口调用都必须在 config 接口获得结果之后,
93         *config 是一个客户端的异步操作,所以如果需要在页面加载时就调用相关接口,
94         *则需要把相关接口放在 ready 函数中调用来确保正确执行。对于用户触发时才调用
95         *的接口,则可以直接调用,不需要放在 ready 函数中。
96         */
97
98
99         wx.ready(function(){
100             that.isReady = true;
101             console.log("初始化成功");
102
103             if(that.readySuccessCall.length > 0) {//成功初始化后,执行的事务
104                 $.each(that.readySuccessCall, function(i, n){
105                     n();
106                 });
107             }
108
109             call && call();
110         });
111         /*config 信息验证失败会执行 error 函数,如签名过期导致验证失败,
112         *具体错误信息可以打开 config 的 debug 模式查看,也可以在返回的 res 参数中查看,
113         * 对于 SPA 可以在这里更新签名。
114         */
115         wx.error(function(res){
116             that.isReady = "false";
117             errorCall && errorCall();
118             that.errorSuccessCall && that.errorSuccessCall();
119         });

```

```
120     }  
121 }
```

【代码解释】

- 本例中引入了腾讯地图的 API，官方参考地址为：http://lbs.qq.com/javascript_v2/demo.html。
- 由于微信 JSSDK 的后台接口不稳定，如果遇到微信 JSSDK 不能用的情况，请清理缓存文件。
- 本例中的代码逻辑首先是检测用户的客户端是否在微信中，优先使用微信的定位服务，如果不在微信中，再使用腾讯地图的定位服务。
- 本例主要用到了微信 JSSDK 的定位服务，以及腾讯地图的反地址解析服务，只要按照 API 的约定传入经纬度就可以使用。
- HTML 5 定位与纠偏是腾讯地图针对最新的移动浏览器提供的服务，目前大部分浏览器都支持，但是 H5 本身的定位有偏差，因此需要腾讯的 API 进行纠正。

12.3 案例总结

本章让读者们接触了 GPS 的一些课题，在实际的项目中，可能遇到更为复杂的案例，例如国际化的项目，那就需要针对不同的国家，使用不同的服务提供商的 API，目前世界范围内流行的是谷歌地图。每个国家、地区都有自己的政策，具体使用什么服务，取决于你的用户在哪个国家、哪个区域。

第 13 章 微信 JSSDK 相关开发

技巧及问题



在日常的开发实践中，使用微信 JSSDK 的时候，会遇见各种各样的问题，由于微信 JSSDK 的公测时间比较短，好多 bug 都无法避免，有的是微信 JSSDK 所致，有的则不是微信 JSSDK 所致，所以需要开发者自己有能力分析、解决这些问题，并且在写代码的同时运用一些小技巧，提高开发效率。

13.1 判断微信浏览器

本书的主题与微信有关，第一个任务就是要判断当前的浏览器是否为微信浏览器，判断方式分为前端与后台。

前端方案主要是访问“Navigator”对象，这个对象包含有关浏览器的信息，其中属性“navigator.userAgent”声明了浏览器用于 HTTP 请求的用户代理头，是一个只读的字符串。识别微信浏览器的相关信息被标识在这个字符串中。代码如下：

```
01  <!DOCTYPE html>
02  <html lang="en">
03  <head>
04      <meta charset="UTF-8">
05      <title>第 13 章 13.1 节</title>
06  </head>
07  <body>
08      <h1 style="font-size: 40px">.</h1>
```



```

09 <b style="font-size: 20px">微信浏览器判断 Demo! </b>
10 <script type="text/javascript">
11     //直接写演示代码
12     var ua = navigator.userAgent.toLowerCase();//获取浏览器标示字符
13     if(ua.match(/MicroMessenger/i)=="micromessenger") { //检测是否含有微信标示
14         alert("我是微信浏览器");
15     } else {
16         alert("我不是微信浏览器");
17     }
18 </script>
19 </body>
20 </html>

```

【代码解释】

- 微信的标识符是“micromessenger”，通过正则检测浏览器字符中是否含有此标识符即可。
- “match”方法可以在字符串内检索特定的值，会找到一个或多个与正则匹配的字符。它必须有两个参数，第一个参数是需要检测的值，第二个是匹配模式的 RegExp 对象。

13.2 微信浏览器不支持打开 App Store 页面

在目前的微信浏览器中，有的版本不支持 APP Store 页面，估计是微信浏览器有什么限制或者有什么利益冲突，但是以腾讯的开放程度来看没必要限制。

例如在页面中写个如下的代码：

```
01 href="https://itunes.apple.com/cn/app/wo-ai/id955760733?mt=8">download</a>
```

以上代码，在微信浏览器内点击的时候，会出现没有任何响应的 bug，但是如果此地址变为 360、百度等链接地址，那么就会没问题。

解决这个问题的主要方案就是在微信浏览器内，利用微信的 API。

第 1 步，判断浏览器是否为微信浏览器，JS 代码如下：

```
01 var isWechat = function () {
```

```

02     var ua = navigator.userAgent.toLowerCase();
03     if(ua.match(/MicroMessenger/i)=="micromessenger") {
04         return true;
05     } else {
06         return false;
07     }
08 }

```

当然，你也可以采用上一节介绍的后台语言（PHP、Ruby、Node.js 等）判断浏览器是否为微信浏览器。

第 2 步，将原来的链接利用“encodeURIComponent”编码，结合“http://mp.weixin.qq.com/mp/redirect?url=”生成新的链接地址，结果如下：

```

01 "http://mp.weixin.qq.com/mp/redirect?url=https%3A%2F%2Fitunes.apple.com%2Fcn%2Fapp
02 %2Fwo-ai%2Fid955760733%3Fmt%3D8"

```

第 3 步，完整的示例代码方法如下：

```

01 <!DOCTYPE html>
02 <html lang="en">
03 <head>
04     <meta charset="UTF-8">
05     <title>第 13 章 13.2 节</title>
06 </head>
07 <body>
08 <h1 style="font-size: 40px">.</h1>
09 <b style="font-size: 20px">App Store Demo! </b>
10 <p style="font-size: 20px" id="conversionUrl"></p>
11 <script type="text/javascript">
12     var isWechat = function(){
13         var ua = navigator.userAgent.toLowerCase(); //获取浏览器字符
14         if(ua.match(/MicroMessenger/i)=="micromessenger"){ //判断是否为微信浏览器
15             return true;
16         } else {
17             return false;
18         }

```

```

19         },
20         conversionUrl = function(url){           //根据浏览器类型转换 url
21             if(isWechat()){
22                 return
23                 "http://mp.weixin.qq.com/mp/redirect?url="+encodeURIComponent(url);//转换为微信内可用的
24                 地址
25             }else{
26                 return url;                       //不转换
27             }
28         },
29         _url =
30         conversionUrl("https://itunes.apple.com/cn/app/wo-ai/id955760733?mt=8");
31         window.onload = function(){               //页面载入成功后修改 dom
32             document.getElementById("conversionUrl").innerHTML = "转换之后的 url: "+_url;
33         }
34     </script>
35 </body>
36 </html>

```

13.3 一些提示码的问题

很多刚玩微信 JSSDK 开发的朋友对微信的一些提示码和一些问题可能还不熟悉，那么本节简单介绍一下。

微信 JSSDK 为开发者提供了调试模式与非调试模式，主要是在“config”配置的时候通过传入的参数“debug”等于“true”或“false”来改变模式。在调试模式下，页面会调用“alert”弹出错信息窗口，下面就列出一些常见的问题及解决方法。

13.3.1 提示码：invalid url domain

原因：域名与 appid 没有绑定成功。

建议：微信 JSSDK 的使用，需要将当前的域名与 appid 进行绑定，如果有端口号，则绑定的时候也要填写端口号，目前一个 appid 仅支持 3 个安全域名绑定。

13.3.2 提示码: invalid signature

原因: 签名错误导致。

建议: 重新检测一下签名顺序及算法。

- 确认签名算法是否正确, 可以使用官网工具 <http://mp.weixin.qq.com/debug/cgi-bin/sandbox?t=jsapisign> 进行校验。
- 检查“config”中的“nonceStr”(JS 中的驼峰标准大写 S), 确认其中的“timestamp”与签名中的“noncestr”是否对应一致。
- 确保配置选项中的“appid”与“jsapi_ticket”的一致性。
- 确保缓存的“access_token”和“jsapi_ticket”是在正常使用的时间周期之内。
- 确保 url 页面路由的完整性及动态性(当前页面), 前端需要使用 JS 获取除去“#”hash 部分的链接, 例如使用如下代码:

```
01 location.href.split('#')[0]; //获取链接
```

13.3.3 提示码: the permission value is offline verifying

原因: “config”没有正确地配置、执行或调用的 API 没有被配置在 config 的 ApiList 参数中(有时候会产生不提示此错误, API 没反应的现象)。

建议: 首先检查一下“config”是否正确, 然后查看是否是在微信 JSSDK 初始化完毕后执行的程序, 如果不是, 则在“wx.ready”回调中执行, 最后确认 jsApiList 的参数中是否存在待使用的 API。

13.3.4 提示码: permission denied

原因: 没有权限使用此 API 或配置的 API 没有传入。

建议: 升级公众号的相关权限或检查 API 参数是否正确。

13.3.5 提示码: function not exist

原因: 当前客户端版本不支持。

建议: 升级客户端, 如果用户在使用你的项目, 那么需要有一个良好的引导升级过程。

附录 A JS-SDK 使用权限

签名算法

A.1 jsapi_ticket

生成签名之前必须先了解一下 jsapi_ticket，jsapi_ticket 是公众号用于调用微信 JS 接口的临时票据。正常情况下，jsapi_ticket 的有效期为 7200 秒，通过 access_token 来获取。由于获取 jsapi_ticket 的 API 调用次数非常有限，频繁刷新 jsapi_ticket 会导致 API 调用受限，影响自身业务，开发者必须在自己的服务全局中缓存 jsapi_ticket。

A.1.1 获取 access_token

access_token 是公众号的全局唯一票据，公众号调用各接口时都需要使用 access_token，开发者需要进行妥善保存。access_token 的存储至少要 512 个字符。access_token 的有效期限目前为 2 个小时，需定时刷新，重复获取将导致上次获取的 access_token 失效。

公众平台的 API 调用所需的 access_token 的使用及生成方式说明：

1. 为了保密 appsecret，第三方需要一个用于 access_token 获取和刷新的中控服务器。而其他业务逻辑服务器所使用的 access_token 均来自于该中控服务器，不应该各自去刷新，否则会造成 access_token 被覆盖而影响业务；

2. 目前 access_token 的有效期限通过返回的 expire_in 来传达，目前是 7200 秒之内的值。中控服务器需要根据这个有效时间提前去刷新 access_token。在刷新过程中，中控服务器对外输出的依然是老的 access_token，此时公众平台后台会保证在刷新短时间内，新老 access_token 都可用，这保证了第三方业务的平滑过渡；

3. `access_token` 的有效时间可能会在未来有所调整, 所以中控服务器不仅需要内部定时主动刷新, 还需要提供被动刷新 `access_token` 的接口, 这样便于业务服务器在 API 调用获知 `access_token` 已超时的情况下, 可以触发 `access_token` 的刷新流程。

A.1.2 获取 `jsapi_ticket`

用上节拿到的 `access_token`, 采用 http GET 方式请求获得 `jsapi_ticket` (有效期 7200 秒, 开发者必须在自己的服务全局缓存 `jsapi_ticket`): `https://api.weixin.qq.com/cgi-bin/ticket/getticket?access_token=ACCESS_TOKEN&type=jsapi` 成功返回如下 JSON:

```
{
  "errcode": 0,
  "errmsg": "ok",
  "ticket": "bXLdikRXVbTPdHSM05e5u5sUoXNKd8-41ZO3MhKoyN5OfkWITDGgnr2fwJ0m9E8NYzWKVZvdVtaUgWvdsdshFKA",
  "expires_in": 7200
}
```

获得 `jsapi_ticket` 就可以生成 JS-SDK 权限验证的签名了。

A.2 签名算法

签名生成规则如下: 参与签名的字段包括 `noncestr` (随机字符串)、有效的 `jsapi_ticket`、`timestamp` (时间戳)、`url` (当前网页的 URL, 不包含 # 及其后面部分); 对所有待签名参数按照字段名的 ASCII 码从小到大排序 (字典序); 使用 URL 键值对的格式 (即 `key1=value1&key2=value2...`); 拼接成字符串 `string1`。

注意: 所有参数名均为小写字母。对 `string1` 作 sha1 加密, 字段名和字段值都采用原始值, 不进行 URL 转义。

即 `signature=sha1(string1)`。示例如下:

```
noncestr=Wm3WZYTPz0wzccnW
jsapi_ticket=sM4AOvdWfPE4DxkXGEs8VMCPGGVi4C3VM0P37wVUCFvkVAY_90u5h9nbSIYy3-SI-HhTdf12fzFy1AOcHKP7qg
timestamp=1414587457
```

```
url=http://mp.weixin.qq.com?params=value
```

1. 对所有待签名参数按照字段名的 ASCII 码从小到大排序（字典序）后，使用 URL 键值对的格式（即 key1=value1&key2=value2...），最后拼接成字符串 string1:

```
jsapi_ticket=sM4AOVdWfPE4DxkXGEs8VMCPGGVi4C3VM0P37wUUCFvkVAy_90u5h9nbSIYy3-SI-HhTdfI2fzFy1AOcHKP7qg&noncestr=Wm3WZYTPz0wzccnW&timestamp=1414587457&url=http://mp.weixin.qq.com?params=value
```

2. 对 string1 进行 sha1 签名，得到 signature:

```
0f9de62fce790f9a083d5c99e95740ceb90c27ed
```

注意:

- 签名用的 noncestr 和 timestamp 必须与 wx.config 中的 nonceStr 和 timestamp 相同。
- 签名用的 url 必须是调用 JS 接口页面的完整 URL。
- 出于安全考虑，开发者必须在服务器端实现签名的逻辑。

附录 B 所有 JS 接口列表



下面是所有 JS 接口列表的简单说明：

1. onMenuShareTimeline

获取“分享到朋友圈”按钮点击状态及自定义分享内容接口。实例及注释如下：

```
wx.onMenuShareTimeline({
  title: "",      // 分享标题
  link: "",       // 分享链接
  imgUrl: "",     // 分享图标
  success: function () {
    // 用户确认分享后执行的回调函数
  },
  cancel: function () {
    // 用户取消分享后执行的回调函数
  }
});
```

2. onMenuShareAppMessage

获取“分享给朋友”按钮点击状态及自定义分享内容接口。实例及注释如下：

```
wx.onMenuShareAppMessage({
  title: "",      // 分享标题
  desc: "",       // 分享描述
  link: "",       // 分享链接
  imgUrl: "",     // 分享图标
  type: "",       // 分享类型，music、video 或 link，不填默认为 link
```



```

dataUrl: "", // 如果 type 是 music 或 video，则要提供数据链接，默认为空
success: function () {
    // 用户确认分享后执行的回调函数
},
cancel: function () {
    // 用户取消分享后执行的回调函数
}
});

```

3. onMenuShareQQ

获取“分享到 QQ”按钮点击状态及自定义分享内容接口。实例及注释如下：

```

wx.onMenuShareQQ({
    title: "", // 分享标题
    desc: "", // 分享描述
    link: "", // 分享链接
    imgUrl: "" // 分享图标
    success: function () {
        // 用户确认分享后执行的回调函数
    },
    cancel: function () {
        // 用户取消分享后执行的回调函数
    }
});

```

4. onMenuShareWeibo

获取“分享到腾讯微博”按钮点击状态及自定义分享内容接口。实例及注释如下：

```

wx.onMenuShareWeibo({
    title: "", // 分享标题
    desc: "", // 分享描述
    link: "", // 分享链接
    imgUrl: "" // 分享图标
    success: function () {
        // 用户确认分享后执行的回调函数
    },

```

```
cancel: function () {
  // 用户取消分享后执行的回调函数
}
});
```

5. onMenuShareQZone

获取“分享到 QQ 空间”按钮点击状态及自定义分享内容接口。实例及注释如下：

```
wx.onMenuShareQZone({
  title: "",    // 分享标题
  desc: "",    // 分享描述
  link: "",    // 分享链接
  imgUrl: "",  // 分享图标
  success: function () {
    // 用户确认分享后执行的回调函数
  },
  cancel: function () {
    // 用户取消分享后执行的回调函数
  }
});
```

6. startRecord

开始录音接口。实例如下：

```
wx.startRecord();
```

7. stopRecord

停止录音接口。实例如下：

```
wx.stopRecord({
  success: function (res) {
    var localId = res.localId;
  }
});
```

8. onVoiceRecordEnd

监听录音自动停止接口。实例及注释如下：

```
wx.onVoiceRecordEnd({  
  // 录音时间超过一分钟没有停止的时候会执行 complete 回调  
  complete: function (res) {  
    var localId = res.localId;  
  }  
});
```

9. playVoice

播放语音接口，实例及注释如下：

```
wx.playVoice({  
  localId: "  // 需要播放的音频的本地 ID，由 stopRecord 接口获得  
});
```

10. pauseVoice

暂停播放接口，实例及注释如下：

```
wx.pauseVoice({  
  localId: "  // 需要暂停的音频的本地 ID，由 stopRecord 接口获得  
});
```

11. stopVoice

停止播放接口，实例及注释如下：

```
wx.stopVoice({  
  localId: "  // 需要停止的音频的本地 ID，由 stopRecord 接口获得  
});
```

12. onVoicePlayEnd

监听语音播放完毕接口，实例及注释如下：

```
wx.onVoicePlayEnd({  
  serverId: "",  // 需要下载的音频的服务器端 ID，由 uploadVoice 接口获得  
  success: function (res) {
```

```

        var localId = res.localId;    // 返回音频的本地 ID
    }
});

```

13. uploadVoice

上传语音接口，实例及注释如下：

```

wx.uploadVoice({
  localId: "",                                // 需要上传的音频的本地 ID，由 stopRecord 接口获得
  isShowProgressTips: 1                      // 默认为 1，显示进度提示
  success: function (res) {
    var serverId = res.serverId; // 返回音频的服务器端 ID
  }
});

```

14. downloadVoice

下载语音接口，实例及注释如下：

```

wx.downloadVoice({
  serverId: "",                                // 需要下载的音频的服务器端 ID，由 uploadVoice 接口获得
  isShowProgressTips: 1                      // 默认为 1，显示进度提示
  success: function (res) {
    var localId = res.localId;    // 返回音频的本地 ID
  }
});

```

15. chooseImage

拍照或从手机相册中选图接口，实例及注释如下：

```

wx.chooseImage({
  success: function (res) {
    var localIds = res.localIds; // 返回选定照片的本地 ID 列表，localId 可以作为 img 标签的 src
    属性显示图片
  }
});

```


16. previewImage

预览图片接口，实例及注释如下：

```
wx.previewImage({
  current: "", // 当前显示的图片链接
  urls: []     // 需要预览的图片链接列表
});
```

17. uploadImage

上传图片接口，实例及注释如下：

```
wx.uploadImage({
  localId: "",           // 需要上传的图片的本地 ID，由 chooseImage 接口获得
  isShowProgressTips: 1 // 默认为 1，显示进度提示
  success: function (res) {
    var serverId = res.serverId; // 返回图片的服务器端 ID
  }
});
```

18. downloadImage

下载图片接口，实例及注释如下：

```
wx.downloadImage({
  serverId: "",           // 需要下载的图片的服务器端 ID，由 uploadImage 接口获得
  isShowProgressTips: 1 // 默认为 1，显示进度提示
  success: function (res) {
    var localId = res.localId; // 返回图片下载后的本地 ID
  }
});
```

19. translateVoice

识别音频并返回识别结果接口，实例及注释如下：

```
wx.translateVoice({
  localId: "",           // 需要识别的音频的本地 ID，由录音相关接口获得
  isShowProgressTips: 1, // 默认为 1，显示进度提示
  success: function (res) {
```

```

alert(res.translateResult); // 语音识别的结果
}
});

```

20. getNetworkType

获取网络状态接口，实例及注释如下：

```

wx.getNetworkType({
  success: function (res) {
    var networkType = res.networkType; // 返回网络类型 2G、3G、4G、WiFi
  }
});

```

21. openLocation

使用微信内置地图查看位置接口，实例及注释如下：

```

wx.openLocation({
  latitude: 0,           // 纬度，浮点数，范围为 90 ~ -90
  longitude: 0,          // 经度，浮点数，范围为 180 ~ -180
  name: "",              // 位置名
  address: "",           // 地址详情说明
  scale: 1,              // 地图缩放级别,整形值,范围从 1~28，默认为最大
  infoUrl: "",           // 在查看位置界面底部显示的超链接,可点击跳转
});

```

22. getLocation

获取地理位置接口，实例及注释如下：

```

wx.getLocation({
  timestamp: 0,          // 位置签名时间戳，仅当需要兼容 6.0.2 版本之前时提供
  nonceStr: "",          // 位置签名随机串，仅当需要兼容 6.0.2 版本之前时提供
  addrSign: "",          // 位置签名，仅当需要兼容 6.0.2 版本之前时提供，详见附录 D
  success: function (res) {
    var longitude = res.longitude; // 纬度，浮点数，范围为 90 ~ -90
    var latitude = res.latitude;   // 经度，浮点数，范围为 180 ~ -180
    var speed = res.speed;         // 速度，以米/每秒计
  }
});

```

```
var accuracy = res.accuracy; // 位置精度
}
});
```

23. hideOptionMenu

隐藏右上角菜单接口，实例如下：

```
wx.hideOptionMenu();
```

24. showOptionMenu

显示右上角菜单接口，实例如下：

```
wx.showOptionMenu();
```

25. hideMenuItems

批量隐藏功能按钮接口，实例及注释如下：

```
wx.hideMenuItems({
  menuList: [] // 要隐藏的菜单项，所有 menu 项见附录 C
});
```

26. showMenuItems

批量显示功能按钮接口，实例及注释如下：

```
wx.showMenuItems({
  menuList: [] // 要显示的菜单项，所有 menu 项见附录 C
});
```

27. hideAllNonBaseMenuItem

隐藏所有非基础按钮接口，实例如下：

```
wx.hideAllNonBaseMenuItem();
```

28. showAllNonBaseMenuItem

显示所有功能按钮接口，实例如下：

```
wx.showAllNonBaseMenuItem();
```

29. closeWindow

关闭当前网页窗口接口，实例如下：

```
wx.closeWindow();
```

30. scanQRCode

调起微信扫一扫接口，实例及注释如下：

```
wx.scanQRCode({
  desc: 'scanQRCode desc',
  needResult: 0,           // 默认为 0，扫描结果由微信处理，1 则直接返回扫描结果
  scanType: ["qrCode","barCode"], // 可以指定扫二维码还是一维码，默认二者都有
  success: function () {
    var result = res.resultStr; // 当 needResult 为 1 时，扫码返回的结果
  }
});
```

31. chooseWXPAY

发起一个微信支付请求，实例及注释如下：

```
wx.chooseWXPAY({
  timestamp: 0, // 支付签名时间戳
  noncestr: "", // 支付签名随机串
  package: "", // 订单详情扩展字符串，详见附录 D
  paySign: "", // 支付签名，详见附录 D
});
```

32. openProductSpecificView

跳转微信商品详情页接口，实例及注释如下：

```
wx.openProductSpecificView({
  productId: "", // 商品 id
  viewType: "" // 0.默认值，普通商品详情页 1.扫一扫商品详情页 2.小店商品详情页
});
```

33. addCard

批量添加卡券接口，实例及注释如下：


```

wx.addCard({
  cardList: [{
    cardId: "",
    cardExt: ""
  }],
  // 需要添加的卡券列表
  success: function (res) {
    var cardList = res.cardList; // 添加的卡券列表信息
  }
});

```

34. chooseCard

调起适用于门店的卡券列表并获取用户选择列表，实例及注释如下：

```

wx.chooseCard({
  shopId: "", // 门店 Id
  cardType: "", // 卡券类型
  cardId: "", // 卡券 Id
  timeStamp: 0, // 卡券签名时间戳
  nonceStr: "", // 卡券签名随机串
  cardSign: "", // 卡券签名，详见附录 D
  success: function (res) {
    var cardList = res.cardList; // 用户选中的卡券列表信息
  }
});

```

35. openCard

查看微信卡包中的卡券接口，实例及注释如下：

```

wx.openCard({
  cardList: [{
    cardId: "",
    code: ""
  }],
  // 需要打开的卡券列表
});

```

附录 C 所有菜单项列表 →

1. 基本类

- 举报: "menuItem: exposeArticle"
- 调整字体: "menuItem: setFont"
- 日间模式: "menuItem: dayMode"
- 夜间模式: "menuItem: nightMode"
- 刷新: "menuItem: refresh"
- 查看公众号 (已添加): "menuItem: profile"
- 查看公众号 (未添加): "menuItem: addContact"

2. 传播类

- 分享给朋友: "menuItem: share: appMessage"
- 分享到朋友圈: "menuItem: share: timeline"
- 分享到 QQ: "menuItem: share: qq"
- 分享到 Weibo: "menuItem: share: weiboApp"
- 收藏: "menuItem: favorite"
- 分享到 FB: "menuItem: share: facebook"
- 分享到 QQ 空间: "menuItem: share: QZone"

3. 保护类

- 编辑标签: "menuItem: editTag"
- 删除: "menuItem: delete"
- 复制链接: "menuItem: copyUrl"
- 原网页: "menuItem: originPage"

- 阅读模式: "menuItem: readMode"
- 在 QQ 浏览器中打开: "menuItem: openWithQQBrowser"
- 在 Safari 中打开: "menuItem: openWithSafari"
- 邮件: "menuItem: share: email"
- 一些特殊公众号: "menuItem: share: brand"

附录 D 卡券扩展字段及签名生成算法



卡券签名和 JSSDK 的签名完全独立，两者的算法和意义完全不同，请不要混淆。JSSDK 的签名是使用所有 JS 接口都需要走的一层鉴权，用以标识调用者的身份，和卡券本身并无关系。其次，卡券的签名考虑到协议的扩展性和简单的防数据删改，设计了一套独立的签名协议。另外由于历史原因，卡券的 JS 接口先于 JSSDK 出现，当时的 JSAPI 并没有鉴权体系，所以在卡券的签名里也加上了 appsecret/api_ticket 这些身份信息，希望开发者理解。

D.1 卡券 api_ticket

卡券 api_ticket 是用于调用卡券相关接口的临时票据，有效期为 7200 秒，通过 access_token 来获取。这里要注意与 jsapi_ticket 区分开来。由于获取卡券 api_ticket 的 API 调用次数非常有限，频繁刷新卡券 api_ticket 会导致 API 调用受限，影响自身业务，开发者必须要在自己的服务全局中缓存卡券 api_ticket。

1. 获取 access_token，附录 A 中已经提到详细的步骤，这里省略。
2. 用上一步拿到的 access_token 采用 http GET 方式请求获得卡券 api_ticket（有效期 7200 秒，开发者必须要在自己的服务全局中缓存卡券 api_ticket）：https://api.weixin.qq.com/cgi-bin/ticket/getticket?access_token=ACCESS_TOKEN&type=wx_card。

D.2 卡券扩展字段 cardExt 说明

cardExt 本身是一个 JSON 字符串，是商户为该张卡券分配的唯一性信息，包含如表

D.1 所示的字段。

表D.1

字段	是否必填	说明
code	否	指定的卡券code码，只能被领一次。use_custom_code字段为true的卡券必须填写，非自定义code不必填写。
openid	否	指定领取者的openid，只有该用户能领取。bind_openid字段为true的卡券必须填写，非自定义openid不必填写。
timestamp	是	时间戳，商户生成从1970年1月1日00:00:00至今的秒数，即当前的时间，且最终需要转换为字符串形式，由商户生成后传入。
nonce_str	是	随机字符串，由开发者设置传入，加强签名的安全性，不长于32位，推荐使用大小写字母和数字。
signature	是	签名，商户将接口列表中的参数按照指定方式进行签名，签名方式使用SHA1，具体签名方案参见下文，由商户按照规范签名后传入。

上述列表中的 signature，这里详细做个说明。

signature 是将 api_ticket、timestamp、card_id、code、openid、nonce_str 的 value 值进行字符串的字典序排序。即将所有参数字符串拼接成一个字符串，然后进行 SHA1 加密，最终得到 signature。

signature 中的 timestamp、nonce 字段和 card_ext 中的 timestamp、nonce_str 字段必须保持一致。

假如：

```
code=jonyqin_1434008071, timestamp=1404896688, card_id=pjZ8Yt1XGILfi-FUsewpnnolGgZk,
api_ticket=ojZ8YtyVyr30HheH3CM73y7h4jJE, nonce_str=jonyqin
```

则：

```
signature=sha1(pjZ8Yt1XGILfi-FUsewpnnolGgZkjonyqin_1434008071ojZ8YtyVyr30HheH3CM73y
7h4jJE1404896688jonyqin)=4F76593A4245644FAE4E1BC940F6422A0C3EC03E
```

说明：JSSDK 使用者请注意本附录，JSAPI 用户可以跳过。



欢迎反馈意见或投稿
 邮箱: dongying@phei.com.cn
 电话: 010-88254047
 微信号: yingzidd

微信公众平台 网页开发实战 HTML5+JSSDK混合开发解密

2017年，微信已经全面支持公众平台内的网页开发，并且提供了网页开发样式库、JSSDK库和网页内支付、网页调试工具，Web开发人员可以利用这些功能开发出更多、更好、更炫的微信内应用。本书立志于帮助微信开发者更好地学习微信中的网页开发和JSSDK库，涵盖了微信网页开发的所有API案例实战以及所有样式库。全书分13章：

- 第1~2章介绍了不同类型微信公众号的区别和微信内网页JSSDK的开发环境；
- 第3~9章介绍了微信开放的所有API，并提供了最贴近现实的实例；
- 第10章通过实例演示了微信提供的网页样式；
- 第11~12章利用微信JSSDK+HTML5技术实现了两个大的项目应用；
- 第13章介绍了一些微信开发的技巧及问题。

本书适合所有微信开发人员使用，尤其适合目前从事Web开发工作的人员。本书定位基础，用案例结合API应用，是目前市场上鲜有的介绍微信网页结合HTML5开发的书。



博文视点Broadview



@博文视点Broadview

上架建议：移动开发>微信

ISBN 978-7-121-31363-9



9 787121 313639 >

定价：69.00元



责任编辑：董 英

封面设计：吴海燕